COMMODORE USERS

48355

K

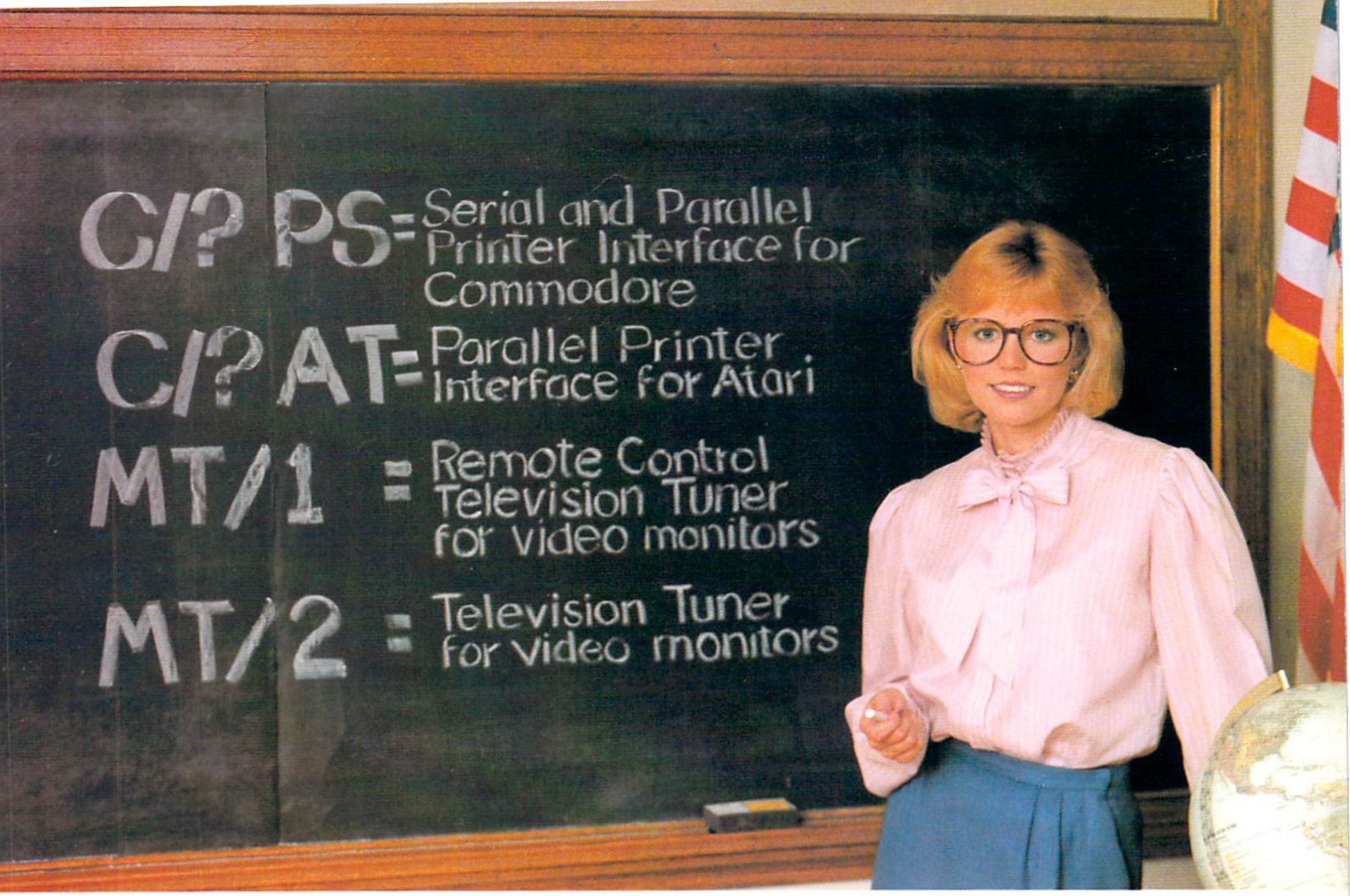
# Ahoy!

$2.50/ CAN.$2.75 DEC. 1984

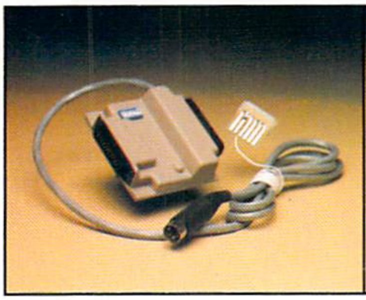



12

0 71896 48355




# Cardco offers a new class . . . of products











**PS- NEW CARDCO PRINTER INTERFACE** for the Commodore-64™ and VIC-20™ Personal Computers. Use with any standard parallel **OR** serial printer. Requires no special programming. Ready to hook-up; includes all necessary cables and plugs. CARD/?PS

**AT- NEW CARDCO PRINTER INTERFACE for the ATARI COMPUTER.** Use your Atari Computer with any standard parallel input printer . . . impact dot matrix, thermal dot matrix, daisy wheel, letter quality, ink jet and laser printer, CARD/?AT includes all necessary cables and connectors.

**MT/1 MONITOR TUNER with REMOTE CONTROL** for any composite color monitor to TURN YOUR MONITOR into a TELEVISION SET. Receive sharper, clearer television pictures on your composite color monitor with a flick of the switch. Separate audio and video outputs; also for stereo sound systems.

**MT/2-UHF/VHF MONITOR TUNER** to turn any composite color monitor into a television set. For color or monochrome monitors; has separate audio and video outputs; receives vivid, bright television pictures that will amaze you.

CARDCO is constantly producing new products. A full line of Commodore hardware and software; letter quality printers for any computer; TV monitor tuners for any composite color monitors and a host of other quality computer products. WRITE for FREE illustrated literature and prices.

**cardco, inc.**
300 S. Topeka
Wichita, Kansas 67202
(316) 267-3807

**"The world's largest manufacturer of Commodore accessories."**



Reader Service No. 43

# Ahoy!

## CONTENTS

## DEPARTMENTS


*A View from the Bridge...of the December issue of* Ahoy! **4**
*Scuttlebutt...some best bets for holiday gift-giving.* **6**
*Reviews...down your chimney—or in your fireplace?* **35**
*Commodares...Dale Rupert issues his year-end challenges.* **55**
*Program Listings...like a stockingful of free software!* **61**


## FEATURES


*A Primer on Printers by Tom Benford* **15**
*Printer Interfacing Revisited by Morton Kevelson* **19**
*A User Guide to KMMM Pascal by Morton Kevelson* **43**
*One Letter is a Thousand Pictures by Orson Scott Card** **47**
*Rupert Report...Dale plunges to the depths of BASIC.*** **86**


*Includes program: *Character Set Example* (64 and VIC)
**Includes program: *Cross Ref* (64 or VIC)

## PROGRAMS


*Construction Co. for the C-64 by B.W. Behling* **58**
*Bug Repellents for the C-64 & VIC by Kleinert & Barron* **62**
*Space Patrol for the C-64 by Bob Lloret* **95**


*Cover illustration by James Regan*
*Illustrations inside screens by Rod Tanabe, John P. Tierney, Young Artists*
*Photo by Ross M. Horowitz*


Publisher
**Michael Schneider**

Editor
**David Allikas**

Managing Editor
**Robert J. Sodaro**

Senior Editor
**Tim Moriarty**

Technical Editor
**David Barron**

Consulting Editors
**Ben Bova**
**Morton Kevelson**
**Dale Rupert**

Art Director
**Raoul Tenazas**

Associate Art Director
**JoAnn Case**

Art Production
**Christopher Carter**
**Eve Griffin**
**Mark Kammerer**

Technical Advisor
**Bob Lloret**

Circulation Director
**W. Charles Squires**

Advertising Director
**Lynne Dominick**

Director of Promotion
**Joyce K. Fuchs**

Controller
**Dan Tunick**

Managing Director
**Richard Stevens**

Advertising Representative
**JE Publishers Representative**
**6855 Santa Monica Blvd., Suite 200**
**Los Angeles, CA 90038**
**(213)467-2266**
**Boston (617)437-7628**
**Dallas (214)660-2253**
**New York (212)724-7767**
**Chicago (312)445-2489**
**Denver (303)595-4331**
**San Francisco (415)864-3252**

AHOY! No. 12, December 1984. Published monthly by Ion International Inc., 45 W. 34th St., Suite 407, New York, NY, 10001. Subscription rate: 12 issues for $19.95, 24 issues for $37.95 (Canada $26.95 and $49.95 respectively). Application to mail at 2nd Class postage rates is pending at New York, NY and additional mailing offices.



Postmaster, send address changes to *Ahoy!*, 45 W. 34th Street, Suite 407, New York, NY 10001. Direct all address changes or matters concerning your subscriptions to *Ahoy!*, P.O. Box #341, Mt. Morris, IL 61054. All editorial inquiries and software and hardware to be reviewed should be sent to 45 W. 34th St., Suite 407, New York, NY 10001.

# VIEW FROM THE BRIDGE

We really *do* put these issues together with a plan in mind. For the December *Ahoy!* you hold in your hands, our plan was to spotlight the peripheral for which the maximum number of our readers would be writing to the north pole. Figuring that most of you already owned a monitor of some kind and a disk drive, the peripheral that seemed to fit the bill was a printer.

Two articles in this issue devote themselves to that theme. Tom Benford's *A Primer on Printers* educates you in the details, both elementary and esoteric, that you'll need to make an intelligent selection. (Turn to page 15.) And Morton Kevelson's *Printer Interfacing Revisited* (part I in this issue) provides the much-requested update to the tutorial/review that appeared in the March and April *Ahoy!* (Turn to page 19.)

Compilers do more than generate high-level programming languages. Every time we've featured them in *Ahoy!* they've generated a great deal of interest. This issue, Morton Kevelson provides *A User Guide to KMMM Pascal,* which will turn your BASIC into the highly structured language of educators. (Turn to page 43.)



*"The computer checks out okay. I don't know what the problem could be."*

Where can you build a house today for under 64K? B.W. Behling will let you do it on the Commodore 64. In fact, he'll let you build a whole block of them, as you manage money, materials, and workers in a race to meet your deadline. This exciting simulation is the next best thing to owning your own *Construction Co.*! (Turn to page 58.)

Fans of Bob Lloret's graphics: the programmer of *Post Time* and *Air Assault* has outdone himself! But don't stare too long at the scrolling, star-filled background of *Space Patrol*...the evil Captain Microz has no appreciation for art! (Turn to page 95.)

If you're at the stage where you consider punching in other people's programs beneath you, try this month's installment of *Creating Your Own Games on the VIC and 64.* Orson Scott Card focuses this time on generating fast graphics with custom character sets. (Turn to page 47.)

Dale Rupert examines the details of program storage by *Diving into BASIC* deeper than you've ever gone! (Turn to page 86.) When you come up for air, try Dale's *Commodares.* (Turn to page 55.)

As always, our reviewers call 'em as they see 'em with regard to releases like *Trivia Fever, Write Now!, Cell Defense,* Mirage's *Database Manager,* and *Gridtrap.* (Turn to page 35.) *Scuttlebutt* fills you in on the most recently announced products for 64 and VIC users, including new printer interfaces, Biblical software, and a host of new games from companies like Activision, Infocom, Victory, HesWare, and Creative Software. (Turn to page 6.)

If you missed our last issue, please note:

1) We've begun utilizing a new method of printing our program listings. Read page 61 of this issue carefully before entering any programs!

2) The *Ahoy!* Bulletin Board is open 24 hours a day, 7 days a week. If you have a 64 or VIC and a modem, call 212-564-7727 to download files providing information on upcoming issues, excerpts from future editions of *Scuttlebutt,* and more.

3) The programs in each month's issue of *Ahoy!* can now be purchased on disk. See page 66.

Computer magazines seldom make room for sentiment—most, if they did, would wind up bumping it for a quarter-page ad. But the relationship between *Ahoy!* and its readers has always been different. We feel as if we know you all from your letters, irate phone calls with programming problems, and now our BBS. That's why we feel that we're talking to close friends when we wish you the best of the holiday season. Please join us again in January, when we begin our second year of growing together in programming knowledge. ***—David Allikas***

# SCUTTLEBUTT

*ANTENNA REDUCER • COMAL PAK • PLUS/4 SOFTWARE • DISK DRIVE AND PERIPHERALS FANS • VOICE MODULE • PRINTER INTERFACES • TERMINAL GRAPHICS • 15-UTILITY DISK • MEMBRANE KEYBOARD • BIBLICAL SOFTWARE • GAMES FROM ACTIVISION, INFOCOM, CREATIVE*

**Looking for a holiday gift for a Commodore owner (like yourself)? *Scuttlebutt* is a great place to browse. One warning though: if you plan to order by mail, check with the company first. Some won't accept mail orders. Some won't deliver in time for Christmas. And some products reported on in *Scuttlebutt* may not be available yet.**



*Two-octave ColorTone Keyboard lets beginners play in correct key and tempo.*
*READER SERVICE NO. 10*

## *MUSICAL KEYBOARD*

Waveform's ColorTone Keyboard, previewed last issue in *Playing the Commodore 64*, has been released. The touch-sensitive membrane includes 25 keys, a touch strip (played like a harp), and 14 function keys. The user can choose from among eight different instruments, and play along with 12 different background songs in one of 12 scales.

The keyboard runs on the 64, independently, with its included software, or with Waveform's *MusiCalc 1*. Retail price is $79.95.

Waveform Corporation, 1912 Bonita Way, Berkeley, CA 94704 (phone: 415-841-9866).

## *INTERFACE FACTS*

Next month, Morton Kevelson continues *Printer Interfacing Revisited* with a look at several available choices. Released too late to be included are the following pair from Omnitronix:

The Deluxe RS232 Interface provides VIC, 64, and SX64 users with all RS232 signals including ring detect. Three switches select for DTE/DCE, pin 5 or pin 20 handshaking, and BUSY line polarity, eliminating the need for rewiring. When supplied with a PCB mounting Female DB25, the unit can substitute for the Commodore 1011A. Price is $39.95.

The Parallel Printer Interface and its driver software allow VIC and 64 users to print Commodore graphics and control codes on practically any dot matrix printer with bit map graphics capability (on printers without, the graphics and codes translate into mnemonics or CHR$ codes). Also $39.95.

Omnitronix Incorporated, P.O. Box 43, Mercer Island, WA 98040 (phone: 206-236-2983).

Seeking a stronger toe hold in the interface arena is the Grappler CD, its price having been lowered by Orange Micro from $139.00 to $119.00. The interface offers four operating modes, built-in screen dump commands, and the ability to render graphics characters on a number of printers. Also available are text mode (for word processing), program listing mode (for transition of control commands), and transparent mode (for access to special printer features).

Orange Micro Inc., 1400 N. Lakeview Ave., Anaheim, CA 92807 (phone: 714-779-2772).




*Has many text formatting features.*
*READER SERVICE NO. 11*

# Flight Simulator II

**For the Commodore 64™**



Put yourself in the pilot's seat of a Piper 181 Cherokee Archer for an awe-inspiring flight over realistic scenery from New York to Los Angeles. High speed color-filled 3D graphics will give you a beautiful panoramic view as you practice takeoffs, landings, and aerobatics. Complete documentation will get you airborne quickly even if you've never flown before. When you think you're ready, you can play the World War I Ace aerial battle game. Flight Simulator II features include ■ animated color 3D graphics ■ day, dusk, and night flying modes ■ over 80 airports in four scenery areas: New York, Chicago, Los Angeles, Seattle, with additional scenery areas available ■ user-variable weather, from clear blue skies to grey cloudy conditions ■ complete flight instrumentation ■ VOR, ILS, ADF, and DME radio equipped ■ navigation facilities and course plotting ■ World War I Ace aerial battle game ■ complete information manual and flight handbook.

## *See your dealer . . .*

or for direct orders enclose $49.95 plus $2.00 for shipping and specify UPS or first class mail delivery. American Express, Diner's Club, MasterCard, and Visa accepted.

**Order Line: 800 / 637-4983**

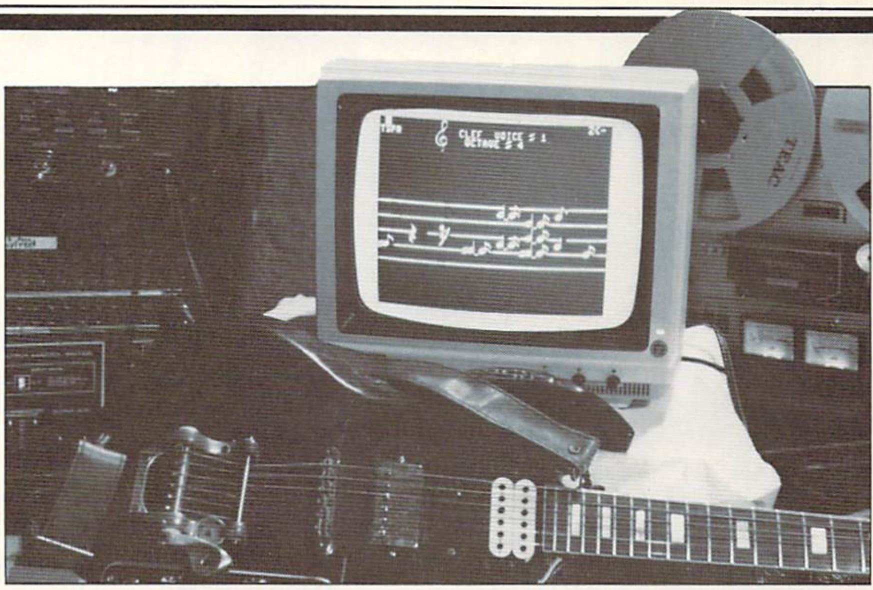
**subLOGIC Corporation**
713 Edgebrook Drive
Champaign IL 61820
(217) 359-8482 Telex: 206995

Reader Service No. 55

## *JAZZED UP*

Entech's *Studio 64* has been enhanced to include hi-res graphics, truer musical notation including tied notes, sharps, and flats, and 12 sample songs including *Billie Jean* and *Sweet Dreams.*

C-64 disk is $39.95, or mail your old disk plus $10 to EnTech Software, P.O. Box 185, Sun Valley, CA 91353 (phone: 818-768-6646).



*New* **Studio 64***'s control key functions facilitate voice and clef changes.*
*READER SERVICE NO. 12*

## *ML TERMINAL GRAPHICS*

Because no BASIC program for transmitting data to graphics terminals can keep up with a 300 baud rate, *Graph-Term 64* is written in 100% machine language. The program lets the user view Tektronix format plots generated by a mainframe computer, download text or plot files, generate plot files on the 64, preview plots on the hi-res screen, and create hard copies of the plots on the Commodore 1520 plotter.

$49.95 plus $4.00 shipping from Bennett Software Company, 3465 Yellowstone Drive, Ann Arbor, MI 48105.

## *LOADED DISK*

So what if you never use some of the 15 utilities found on Applied Technologies' *Bits and Pieces* disk? In these days of bloated software prices, mere satisfaction at getting the programs for $2.00 ($29.95 ÷ 15) each should make their purchase worthwhile.

Various programs allow 64 users to check for errors on a whole disk or by track and sector, examine the contents of a block, a file, or the disk directory, transfer files in various ways, create sprites, and more.

Applied Technologies, Inc., Computer Products Division, Lyndon Way, Kittery, Maine 03904 (phone: 207-439-5074).

## *AS THE CROW FILES*

At first we figured that Audiogenic, Ltd. of England had titled their new database manager *Magpie* because all the good names were taken. Then we remembered that "magpie" can mean a collector of odds and ends. Get it?

23 pop-up menus provide forms design, peripheral control and programmed applications, and use of dual drives and enhanced printers. Record size runs from 25 to 6100 characters, with 26 alpha and 26 variable fields. Disk (for applications use) is $64.95; cartridge (for user-programmed solutions) is $99.95.

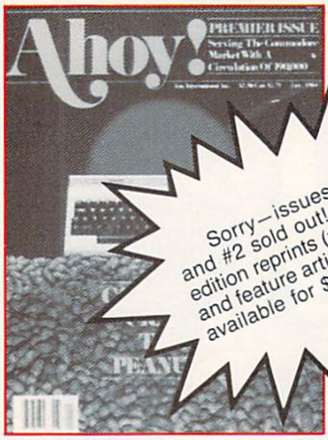
Audiogenic, Ltd., c/o Regenics, Inc., P.O. Box 767, Orange, CA 92666 (phone: 714-639-9396).

## *PLUS/4 PLUS 64*

C-64 users thinking of stepping up eventually to the Plus/4, keep your fingers crossed that other software manufacturers follow the precedent set by Tri Micro. The developers of the Plus/4's integrated software have begun releasing utility programs on two-sided disks: one side for the Plus/4, one for the 64.

*Plus Graph* enhances the graphics capabilities of the Plus/4, generating hi-res graphs prepared within the computer's spreadsheet or database. The 64 version

# Back Issues of Ahoy!





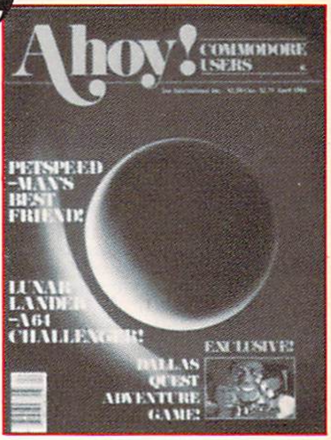


**ISSUE #1–JAN. '84 $4.00**
64 v. the Peanut! Telecommunications! Tales of the Commodore! And ready to enter: Multi Dräw 64 Graphics System! Interrupt Music Maker/Editor! A Peek at Memory! Sequential Files!

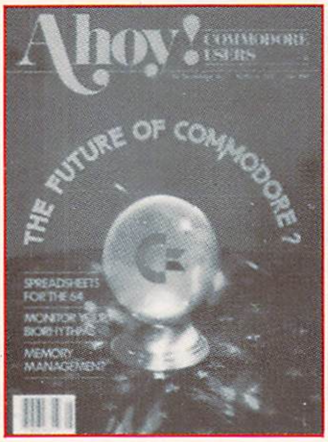
**ISSUE #2–FEB. '84 $4.00**
Illustrated tour of the 1541! Artificial intelligence! Synapse's Ihor Wolosenko interviewed! And ready to enter: Music Maker Part II! Screen Manipulation! Night Attack! Relative Files!



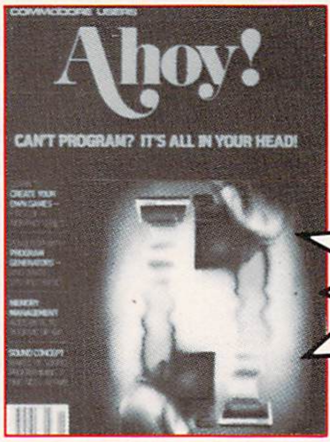

**ISSUE #3–MAR. '84 $4.00**
Anatomy of the 64! Printer interfacing for VIC & 64! Educational software series begins! And ready to enter: Address Book! Space Lanes! Random Files on the 64! Dynamic Power!




**ISSUE #4–APR. '84 $4.00**
Petspeed and Easy Script tutorials! Printer interfacing continues! Laserdisc! And ready to enter: Apple Pie! Lunar Lander! Name that Star! Lower case descenders on the 1525!




**ISSUE #5–MAY '84 $4.00**
Future of Commodore! Inside BASIC storage! Memory management on the VIC & 64! Spreadsheets! And ready to enter: Math Master! Air Assault! Biorhythms! VIC Calculator!




**ISSUE #6–JUNE '84 $4.00**
Game programming column begins! Program generators! Rupert on inputting! Memory management continues! And ready to enter: Post Time for the 64 & VIC! Alpiner! Sound Concept!




**ISSUE #7–JULY '84 $4.00**
MSD Dual Disk! Database buyer's guide! Training your cursor! Screen displays! File Sleuth! Users Groups! And ready to enter: Renumbering! Checklist! Math Defender! Brisk!




**ISSUE #8–AUG. '84 $4.00**
Choosing a word processor! Computational wizardry! Creating your own word games! Sound on the 64! And ready to enter: Micro-Minder! Directory Assistance! Terrible Twins!

**ISSUE #9–SEPT. '84 $4.00**
Program your own text adventure! Build a C-64 cassette interface! Video RAM! And ready to enter: Salvage Diver! DOS! Sound Explorer! Castle of Darkness! Base Conversions!

**ISSUE #10–OCT. '84 $4.00**
C-64 Graphics programs! Bit- mapped graphics! Joystick programming! And ready to enter: VIC 40 Operating System! BAM Read & Print! Emerald Elephant of Cipangu! Lawn Job!

**ISSUE #11–NOV. '84 $4.00**
Music programs & keyboards for the 64! Graphics feature continues! And ready to enter: PTE word processor! Block Editor! Alternate Character Set for the 64! Tunnel of Tomachon!

**If you have a modem and want more complete information on any of the issues listed, call the *Ahoy!* Bulletin Board: 212-564-7727.**

**Try this experiment! Flip through any issue of *Ahoy!* that you've finished reading. Add up the hours of programming toil our feature articles and tutorials saved you—the hours of shopping our expansive news section saved you—and the money our unbiased reviews saved you—and see if the magazine didn't pay for itself! Then ask yourself: can you afford to be without a single time- and money-saving back issue?**

**If you're serious about programming, you can't be serious about continuing without a complete collection of *Ahoy!***

**Send coupon or facsimile to: *Ahoy!* Back Issues, Ion International Inc., 45 West 34th Street—Suite 407, New York, NY 10001.**

Ahoy!

Please Send Me The Following:

____ Copies of issue number ____
____ Copies of issue number ____
____ Copies of issue number ____

Enclosed Please Find My Check or Money Order for $________

(Outside the USA please add $1.00 for every copy)

NAME ____________
ADDRESS ____________
____________
CITY ____________
STATE ____________
ZIP CODE ____________

works with Tri Micro's *The Write File* (integrated word processor and database) and *Your Home Office* (integrated word processor and spreadsheet).

*Your Personal Accountant* keeps track of cash and several bank accounts, providing 21 categories for expenses. No knowledge of bookkeeping is required.

Tri Micro, 1010 N. Batavia, Suite G, Orange, CA 92667 (phone: 714-771-4038).

### *COMMODORE JAMES VERSION*

With the proliferation of Biblical software for the C-64, the Gideons may soon start placing floppy disks in hotel room drawers.

*Knowledge Bible* lets 2-10 players compete to answer questions on the King James Bible. The $39.95 disk contains 500 questions; additional 500-question disks $19.95.

Little David Enterprises, P.O. Box 91, Fairless Hills, PA 19030 (phone: 215-943-0770).

*Scrambled Verses*, one of two from Smoky Mountain Software, provides Scriptural text on various Christian subjects, in random order. To the accompaniment of hi-res pictures and familiar hymns, the player must rearrange the text properly. Disk; $19.95.

*Sort 'Em 1: Creation, The Flood, Baby Moses* displays story pictures of those Old Testament episodes, then shuffles the pictures and requires players aged 5 to adult to put them in order. Also disk, $19.95.

Smoky Mountain Software, 13 Catatoga Path, Brevard, NC 28712 (phone: 704-885-2516).

Of course, Davka Corporation, patriarchs of religious software, are still at it. Their latest, *Search*, simulates a trip to and through the Holy Land by challenging 1-4 players (age 8 to adult) to locate their cousin Sara's house in one of 20 towns in Israel, using maps and various clues. Sara's location changes each time you play. On disk; $24.95.

Davka Corporation, 845 North Michigan Avenue, Suite 843, Chicago, IL 60611 (phone: in IL 312-944-4070; rest of USA 1-800-621-8227).

### *SWAHILI SPOKEN HERE*

They laughed at talking pictures in the beginning. Don't make the same mistake with talking computers. Voice modules for the C-64 are growing in number and capability. One of the more recent (and capable) is Chirpee from Eng Manufacturing. This $179.95 peripheral allows users to perform

a variety of functions via spoken words–in any language on earth, thanks to its phonetic feature basis. All utilities to train, recognize, and store voice files are included and can be copied or inserted into other programs.

Eng Manufacturing, Inc., 4304 W. Saturn Way, Chandler, AZ 85224 (phone: 602-961-0165).

### *OFF THE RAK*

RAK's *Antenna Reducer* will design an antenna 30-100% of full size on your 64. You select the frequency range (.5 to 15MHz) and loading coil diameter; the program then calculates the antenna measurements and graphically displays the antenna design. $7.95 on tape, $10.95 on disk, plus $2.00 shipping.

Also from RAK is *C-64 Lil' Ledger,* allowing the small businessman up to 500 entries in 15 expense and income categories. On disk; $12.95 plus $2.00 shipping.

RAK Electronics, P.O. Box 1585, Orange Park, FL 32067 (phone: 904-264-6777).

### *BIGGEST THING SINCE ESPERANTO?*

COMAL, which combines some of the best features of languages like LOGO, MODULA, PASCAL, and ADA in an easy-to-use format, has crossed the Atlantic. The same COMAL 2.0 cartridge that Commodore is selling in Europe will be distributed here by the Comal Users Group, U.S.A., as will the disk-loaded COMAL 0.14.

Pay attention, now: you can have, for $19.95, the *Enhanced COMAL Pak* (COMAL 0.14, 2 disks, and reference card); for $29.95, the *COMAL Starter Kit* (adds a book, a third disk, and a case); or for $99.95, the *COMAL 2.0 Cartridge Package* (2 books, 2 disks, and the cartridge in a custom molded case). For a stamped and self-addressed envelope you can get a free COMAL Info Pak.

COMAL Users Group, U.S.A., Ltd., 5501 Groveland Ter., Madison, WI 53716 (phone: 608-222-4432).

The Toronto Pet Users Group (TPUG) has published a 50-page COMAL programming manual written by one of the language's creators. Price is $9.95.

TPUG Inc., 1912A Avenue Road, Suite #1, Toronto, Ontario, Canada M5M 4A1 (phone: 416-782-9252).

### *COOL IT*

Does your disk drive tend to be a hot head? Maybe Com-Cool ($54.95) or Com-Cool Plus ($79.95) can cool it down a little bit. These fans sit atop your 1541 and pipe air through the ventilation ducts on top of the unit. With the addition of a special bracket two stacked 1541's can be cooled off. Both units have an air filter to insure that no dirt is blown into your drive. Com-Cool Plus provides a surge protector and two additional sockets to protect your entire computer system from power surges.

City Software, 735 West Wisconsin Ave.—Suite 1010, Milwaukee, WI 53233 (phone: 414-291-5125).

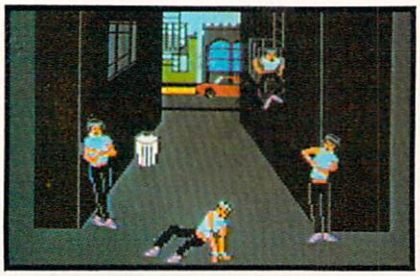
If you have a vacant area measuring 2" by 3.3" by 1.12" inside your printer, monitor, or other peripheral, you can install a Bee Fan. Its two bending piezoceramic mylar blades flap like a bee's wings, using 1/15 the power of rotary fans, or 0.11 of a watt at 120 VAC.

$24.95 plus $3.00 shipping from Atmospheres, 1207 Eighth Avenue, Brooklyn, NY 11215 (phone: 718-788-6799).


THE NEW BREED
Mimic introduces instant evolution!
Meet the Spartan™— the missing link
that turns your Commodore 64™
into a whole new apple.
The Spartan™ now allows you to use
any of the Apple™ II/II+ hardware
and software you want.
Now, **that's** natural selection!
MIMIC
FOR INFORMATION
PLEASE WRITE TO:
MIMIC SYSTEMS INC.
1112 FORT ST., FL. 6F
VICTORIA, B.C.
CANADA V8V 4V2
Commodore 64 and Commodore logo are trademarks of Commodore Electronics Ltd. and/or Commodore Business Machines, Inc.
Apple® II and Apple® II+ are trademarks of Apple Computer Inc.
Spartan™ is a trademark of Mimic Systems Inc., and has no association with Commodore Electronics Ltd. or Apple Computer, Inc.
Reader Service No. 42

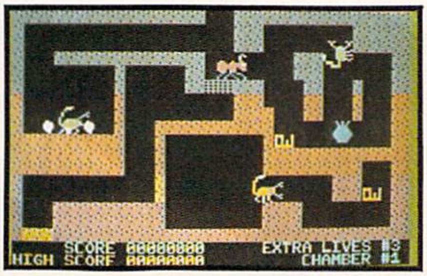
## *NEW GAMES UPDATE*

By the time you read this, we're confident that the only people still moonwalking will be doing it on green cheese. But Creative Software hopes their *Break Street* will succeed despite the ephemeral nature of fads.



*Take a moonwalk down* Break Street.
*READER SERVICE NO. 14*

Guiding your dancer through head spins, snaking, the tut, and other gyrations, you must avoid missing a key sequence. That will cause you to fall and the action to be turned over to the Stingrays, the neighborhood gang you're dancing against. You may string together and save your favorite dance sequences. On C-64 disk; $24.95.

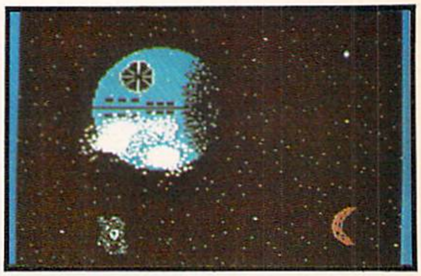



Fire Ant:*an adventure/arcade mix.*
*READER SERVICE NO. 15*

Creative Software, 230 East Caribbean Drive, Sunnyvale, CA 94089 (phone: 408-745-1655).

*Fire Ant,* one of two for the 64 from Victory Software, is an adventure game incorporating joystick-controlled arcade action, eight different screens, and a number of puzzles in need of solving.

*Zeta-7* is a 3D space battle that



*The 3D view from* Zeta-7*'s gunport.*
*READER SERVICE NO. 16*

you view through a gunport as alien ships zoom in from space.

*Continued on page 96*





Reader Service No. 38

# A PRIMER ON PRINTERS

## Photos and Text By Tom Benford

One of the major peripherals you'll purchase for your home computer system is a line printer. Not only is a line printer a highly productive peripheral, it's an expensive one, often costing more than the combined price of the computer and disk drive. So before you lay out your hard-earned cash, it's a good idea to understand what you should look for in the way of features to get the most value for your money. This article will provide you with enough information on printers to make you an "educated consumer" and insure that you'll get a printer that's right for your needs. There are four major types of line printers: dot matrix, letter quality or daisy wheel, thermal, and ink-jet.

Thermal printers utilize a special paper which is heat-sensitive. The print head has wires to form the characters, similar to a dot matrix printer, but instead of striking a ribbon, the wires in the thermal printer heat up and burn off the coating of the special paper. The only advantages of thermal printers are that they're almost silent, and relatively inexpensive. The disadvantages, however, far outweigh these dubious pluses: the specially coated thermal paper is more expensive than plain paper, it is extremely fragile, and it deteriorates over a period of time due to atmospheric and climate conditions. This makes thermal printers highly undesirable for applications where any kind of permanence is desired.

The last of these types, the ink-jet printer, is the product of relatively new technology compared to the former three types, and at this point in time the variety and cost of such units is beyond the scope of this article; we will, however, cover ink-jet printers and color printer/plotters in a subsequent feature in *Ahoy!* For now, we'll concentrate on the dot matrix and letter quality printers.



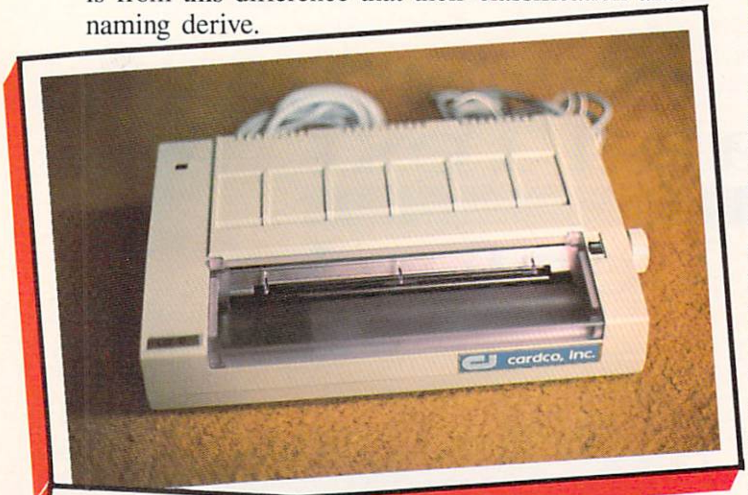




*The MPS-801 offers economical dot matrix printing; the Gemini 15X offers greater versatility at a higher price. See chart on page 18.*
*MPS-801: READER SERVICE NO. 1*
*GEMINI 15X: READER SERVICE NO. 2*

## *DOT MATRIX v. LETTER QUALITY: THE DIFFERENCES*

The major difference between these two genres is the way in which they produce printed characters. It is from this difference that their classification and naming derive.







***The Cardco LQ-2 utilizes a rotating character drum, unlike most letter quality printers (including the LQ-3—bottom).***
***LQ-2: READER SERVICE NO. 3***
***LQ-3: READER SERVICE NO. 4***

In the dot matrix variety, the printed characters are comprised of little dots which form the alphanumeric and graphic characters. These dots are produced by groupings of little wires which form the designated character and strike the ribbon to leave the imprint of the character on the paper. One important asset of dot matrix printers is the ability to produce virtually any character, provided the coding is correct; being able to produce the entire graphics character set of Commodore computers is but one example of this flexibility of character configuration. As with just about everything else in life, there are compromises, and to get this ability, you'll have to give a little as far as readability and character formation go.

The number of dots which are available to comprise the image matrix determines the quality and formation of the characters; the more dots, the better. Often in low-resolution printers, the shapes of characters will be poorly defined and characters with descenders, such as "g", "j", "p", "q", and "y" will look as if they are slightly elevated from the other characters in the word, as the "tail" or descender doesn't dip below the baseline as it should. A 5x7 dot matrix offers lower resolution and print quality than a 7x9 dot matrix. In these examples, the 5x7 matrix has five horizontal rows of dots and 7 vertical rows, yielding a total of 35 (maximum) dots with which to make up the characters. On the other hand, the 7x9 matrix yields 63 dots maximum total, or almost twice as many as the 5x7 matrix. Obviously, you can produce much better character definition and have the descenders formed and positioned correctly when you have this many dots to work with. The old adage "bigger is better" holds true here, with a bigger number of dots in the matrix being a highly desirable feature. Another adage, "better is more expensive," also holds true; high-resolution printers cost more than low-resolution models.

Conversely, letter quality printers have true correspondence-quality output, often rivaling or exceeding the output of the best office typewriters. The daisy wheel is by far the most widely used print mechanism on letter quality printers, although some printers (such as the Cardco LQ-2) utilize rotating drums on which the character matrices are molded. The name daisy wheel comes from the actual physical print device which produces the characters. It's a circular affair with "spokes" radiating out from its hub. On the ends of these "spokes" or "petals" are the individual alphanumeric and punctuation characters that you print from. Overall, it looks somewhat like a daisy, hence the name daisy wheel.

Another difference between the two types of printers is the inking media with which they print. Dot matrix printers usually (but not always) use cloth ribbons impregnated with ink that can often produce hundreds of thousands of characters before they need replacing. They accomplish this by rotating fully in one direction, then reversing direction and reusing the ribbon on the second and subsequent passes.

The daisy wheel or letter quality printer, on the other hand, almost always uses a single-pass mylar ribbon cartridge which must be replaced once it is fully used up. These ribbon cartridges have a much

shorter useful life than the traditional two-spool cloth ribbons, and they cost considerably more. What do you get for the extra expense? Superb print quality that looks as good as or better than the output of an IBM Selectric typewriter.

Print speed is another consideration when contemplating a printer. Dot matrix printers are usually quite fast, with some models exceeding 150 characters per second (cps). Letter quality printers, for the most part, are much slower, but once again, the print quality is much better. It should be noted, however, that there are some high-speed letter quality printers available as well, such as the Diablo 630, but now we're in the $2,000 price range, which is getting away from the main point of this article—how to get the most printer for the least amount of money.

## *BI-DIRECTIONAL PRINTING*

Bi-directional printing is a highly desirable feature to have in a printer, whether it is dot matrix or letter quality. Bi-directional printing simply means that the print head is active when traveling in *both* directions across the paper, rather than mono-directional. The best illustration of mono-directional printing is the ordinary typewriter. The carriage travels from the left edge of the paper to the right, then travels back without printing when the return key is pressed. A bi-directional printer prints from left to right, then the next line is printed from right to left, so the return trip is not wasted. The big advantage of bi-directional printing is speed—about double that of a mono-directional printer.

*Logic-seeking* bi-directional printing is even better, since the printer "looks ahead" in the text to see where the next line will end *in advance.* By knowing in advance where one line ends and the next one begins, the printer is able to move the print head about very fast from one line to the next and another increase in speed is made possible. Bi-directional printers without logic-seeking circuitry print out a line, then travel all the way to the margin before they start to print on the next line. There's a lot of wasted travel here, and this is non-productive time. Not only does it slow things down quite a bit, it puts extra wear on the moving parts of the printer.

## *CARRIAGE WIDTH*

Your intended use for the printer will be the determining factor of how wide a carriage you need. If you plan to print out spreadsheets, for example, you'll need a printer with 132-column or better capability, so you'll need a 15" carriage. On the other hand, if you intend to use the printer primarily for printing out program listings, carrying on correspondence with a word processing program, etc., you'll probably do nicely with a 9" carriage machine. Cost goes up as the carriage width increases, but it may be a good idea to spend the extra money and get a printer with a wide carriage—you may need it eventually. A wide-carriage printer can handle narrower paper as well, but a narrow-carriage printer can't handle wide paper. In the future, you may need that extra carriage width, and you won't have to go out and buy another printer to get it!

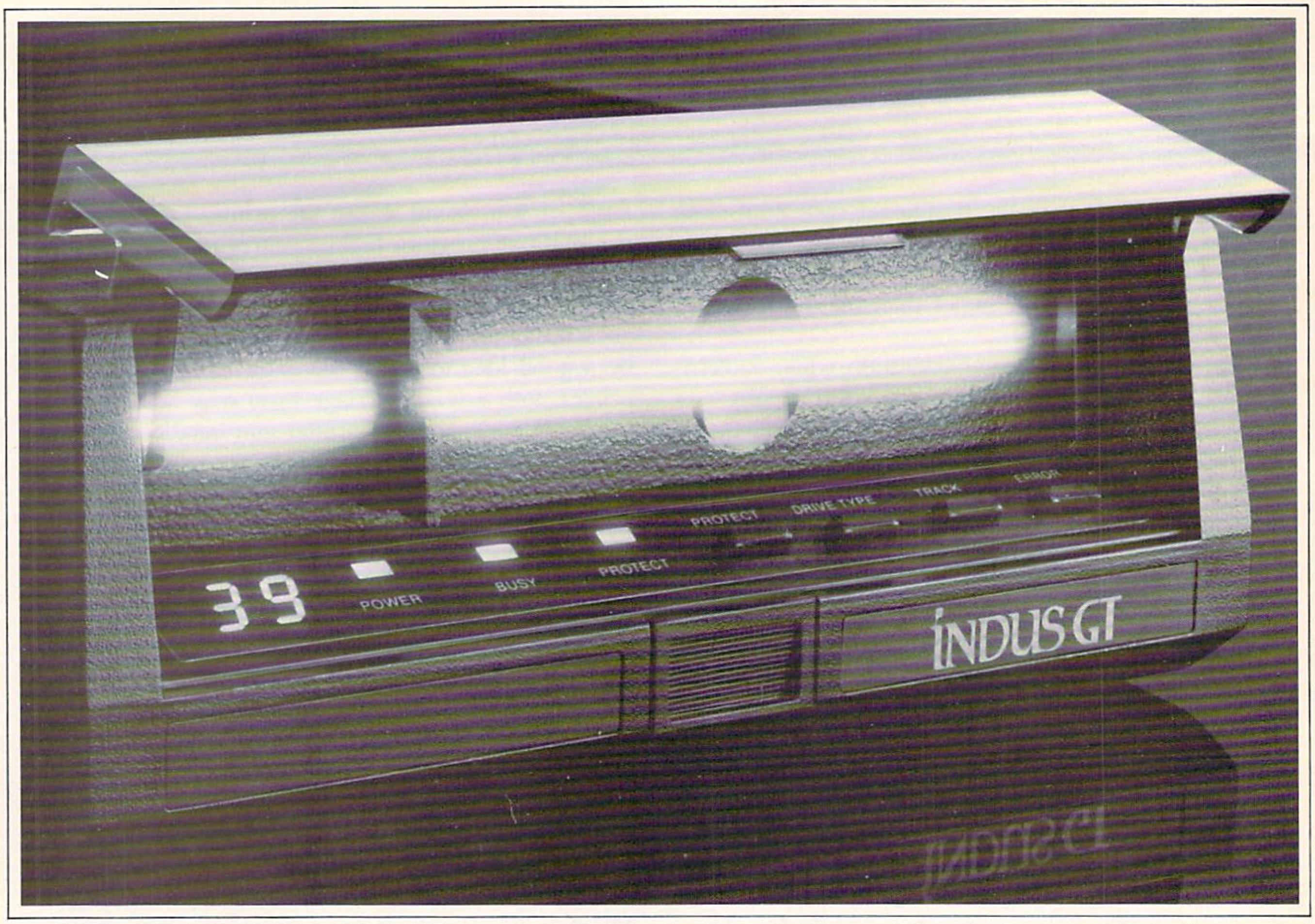
## *FRICTION OR TRACTOR FEED*

Friction, as you know, literally means two objects rubbing together. In a line printer with friction feed, the paper is forced against the platen roller by a pinch roller which causes friction. When the platen roller is rotated, this friction causes the paper to travel with it, and the paper is advanced a la friction, hence the term. A really big plus of friction feed is that it allows you to use single sheets of paper, such as letterhead stock, or continuous-roll paper with equal ease. If the carriage is wide enough, you can even run form-feed tractor paper if you wish.

Cut-sheet feeders are often available on letter quality printers, but they are sometimes offered as optional accessories on dot matrix units as well. A cut-sheet feeder is a hopperlike affair that automatically feeds sheets of cut paper (i.e., letterhead or postcard stock) into the printer. It's a handy device for printing personalized form letters, but often the price of a cut-sheet feeder rivals that of the printer itself! They do offer enhanced speed in printing such material, but you must decide if the extra money involved is worth the increase in speed, and how much of this sort of printing you intend to do should be the deciding factor here.

Tractor feed, as the name implies, utilizes the hole-and-sprocket principle, much like the treads of a tank or the chain drive of old agricultural tractors. The paper has two perforated "tracks" or shoulders, one at each end, with holes in it, spaced at even intervals. The "tractor" section of the printer consists of two gear-driven sprockets with "teeth" or small spikes on them. The idea here is that the teeth or spikes engage in the holes on the shoulder of the paper, and pull it through as the sprockets rotate. The paper has perforations which allow you to tear off the shoulders when printing is completed. Tractor feed is very widely used on printers, and those printers offering friction feed as standard equipment

*Continued on page 92*




# What you get if you cross a Commodore 64 with a Ferrari.

You get the incredible Indus GT™ disk drive.

You get brains. You get beauty. But, that's not all you get.

You get a disk drive that can handle 100% of Commodore's software. 400% faster.

You get the disk drive with the best service record around. With a one year warranty on parts and labor to prove it.

And, you get the only disk drive that comes with free software. Word processing. Spreadsheet. Database manager. Plus, a carrying case that doubles as an 80 disk storage file.

Most of all, you get luxury. From the sleek lines of its soundproofed chassis to the responsive AccuTouch™ controls at the Indus CommandPost.™ From the LED display that keeps you in control of your Commodore to the air-piston operated dust cover that protects your disks and drive.

So, you know what you really get if you cross a Commodore 64 with a Ferrari?

You get the best.




INDUS™



Reader Service No. 45

# PRINTER INTERFACING REVISITED

# PART I:

## The 1525 Printer - A Tutorial

BY MORTON KEVELSON

It is nearly a year since we discussed the interfacing of a non-Commodore printer to the VIC 20 or the Commodore 64 (see our March and April issues). Since that time a number of new and improved printer interfaces have been introduced. We will be taking a close look at just how these products perform in Part II of this feature in our February issue.

The process of reviewing these interfaces, as well as the reader response to our first feature, indicated that printer expertise advanced in three steps:

1) Learning the use of the Commodore 1525 or MPS-801 printer.

2) Learning the features and capabilities of the interface.

3) Learning how to use the features of your particular printer.

The last two steps are dependent on the particular hardware you have purchased. There are too many of these devices for us to go into any great detail for each one. However, the reviews next month should give you some insight into the use of the various printer interfaces.

We can offer two bits of advice. First, *very carefully* read the manual supplied with the equipment. We have found that all the information you will need is in these manuals, although it may not always be obvious. Second, this tutorial on the 1525 printer will be very useful.

One of the primary tasks of the printer interface is to provide total emulation of the Commodore 1525 printer. Thus a knowledge of how to use this printer is helpful. This is particularly important if you will be doing any of your own programming. Trying to drive a non-Commodore printer directly requires translation of Commodore ASCII to standard ASCII. We have found that the manuals which accompany the interfaces are very sketchy on how to use this Commodore product. In addition, the owner of a non-Commodore printer may not have access to a Commodore printer manual. Many of the reader questions we have received can be directly traced to this lack of information. This tutorial will start you in the right direction.

### *THE OPEN STATEMENT*

The computer communicates with all its peripherals via a channel which is specified by the OPEN statement. The syntax is as follows:

```
OPEN fn,dv,sa
```

Notice the three parameters which are associated with this statement. The first, "fn", is the file number, sometimes referred to as the logical file number. This parameter is of interest only to the computer. All information which is transmitted to the printer is routed by the computer via this number. The file number can be any value from 1 to 255, but you should use only numbers less than 128. Numbers greater than 127 have a special meaning to the computer. They will cause it to send an extra line feed with each carriage return. This has the effect of double spacing the text on the printer.

The second number, "dv", is the device number. This is solely determined by the design of the peripheral. Printers are usually device number 4, although 5 is generally possible. Device numbers 8 and above are generally reserved for the disk drive. Other device numbers are 0 for the keyboard, 1 for the cassette deck, 2 for the RS232 port, and 3 for the screen. You must use the proper device number for your printer or interface.

The third number, "sa", is the secondary address, sometimes referred to as the command. This is one of Commodore's ways of passing instructions to a

peripheral device. This number can range from 0 to 255. The particular meaning or action performed by the secondary address is totally dependent on the particular hardware device. If it is omitted, a secondary address of 0 will be assumed.

There are only two possible secondary addresses which have any meaning to the 1525 printer, 0 and 7. A secondary address of 0 sets the printer to "cursor up mode." This means that any valid character data will be printed as either an upper case alphabet text, any of the numbers or standard punctuation symbols, or any of the Commodore graphic symbols which are accessed from the keyboard by pressing the shift key or the Commodore key. Lower case text will not be printed in this mode.

The secondary address of 7 sets the printer to "cursor down mode." This means that any valid character data will be printed as either upper or lower case text, any of the numbers or standard punctuation symbols, or any of the Commodore graphic symbols which are accessed from the keyboard by pressing the Commodore key. The Commodore graphic symbols, which are accessed by pressing the shift key, will not be printed in this mode.

These two secondary addresses actually have the same effect on the printer as the simultaneous pressing of the Commodore and shift keys has on the screen display.

The concept of secondary addresses is very important. Each of the printer interfaces we have seen makes extensive use of the secondary address values which are not used by the 1525 printer. This allows numerous additional printer functions and enhancements to be easily accessed by the user.

## *THE PRINT# STATEMENT*

This statement is used to send text for printing, as well as additional commands, to both the printer and the interface. The syntax is:

```
PRINT#fn,data
```

The value for "fn" is the same file number which was used with the OPEN statement. Take careful note as to how this statement is entered. There should not be any spaces between the PRINT command, the "#" symbol, the file number, and the trailing comma. This is not the same statement as would be used for PRINTing to the screen, although it serves the same function. It cannot be abbreviated with a "?" when writing a program. The actual abbreviation is pR (the letter "p" followed by a shifted "r"). The file number and trailing comma still have to be typed separately.

Aside from all this, the information which follows the statement is handled exactly the same way as for the screen PRINT statement. Thus, semicolons cause continuous printing and commas will tab ten spaces. As with the screen, both of these will also suppress the trailing carriage return. For example,

```
OPEN1,4,7
PRINT#1,"Hi","there"
```

will print

```
Hi        there
```

separated by eight spaces as shown.

## *THE CHR$ FUNCTION*

The 1525 printer can be instructed to perform in several different modes. These are all accessed by sending the printer special codes via the PRINT# statement. This is similar to including screen control codes, such as a clear screen, in a PRINT statement. Since these codes are generally not printable characters, we have to use a special BASIC function to generate them. This is the CHR$ function. The syntax is:

```
CHR$(i)
```

Note the use of the parentheses. The value of "i" can be any integer between 0 and 255, or it can be any BASIC expression which evaluates to this range. The CHR$ function converts the number within the parentheses to a data string which is exactly one byte long. Note that the value which results from a CHR$(1) is not the same as the number 1 or the BASIC string "1". Floating point numbers are actually represented by five data bytes. The string "1" is actually equal to a CHR$(49). Actually, the various printable characters can be used instead of their ASCII codes if you find it more convenient. Many of the printer manuals will represent control codes as printable characters whenever possible. A complete table of Commodore ASCII codes is included in the manual supplied with your computer.

The codes which the 1525 printer can understand are listed in the table on page 21.

Let us take a quick look at each of these codes.

The most simple are the CHR$(145) and CHR$(17) codes. Actually, these perform the identical function as the secondary addresses of 0 and 7 described above. They allow the printer to be switched be-

## TABLE OF 1525 PRINTER COMMAND CODES

| CODE | FUNCTION |
|---|---|
| CHR$(8) | Sets printer to graphic mode |
| CHR$(10) | Sends a line feed without a carriage return |
| CHR$(13) | Carriage return |
| CHR$(14) | Sets printer to double width characters |
| CHR$(15) | Sets printer to standard width characters |
| CHR$(16) | Moves printhead to the specified character or dot position |
| CHR$(17) | Sets printer to cursor down mode |
| CHR$(18) | Turns on reverse field printing |
| CHR$(26) | Graphic repeat command |
| CHR$(27) | Used with CHR$(16) to set dot position |
| CHR$(145) | Sets printer to cursor up mode |
| CHR$(146) | Turns off reverse field printing |

tween the two modes without closing and reopening the channel. They also allow the mixing of upper and lower case text with any of the Commodore graphic symbols on the same printed line.

To obtain proper operation of the CHR$(17) and CHR$(145) codes, the associated text must be sent with the same PRINT# statement. After the PRINT# command is completed, the printer reverts back to the mode specified by the original OPEN statement. This will happen even if the program line is terminated with a comma or a semicolon. Technically speaking, the PRINT# command tells the printer to "listen." After the data is sent, the printer is "unlistened." The following program demonstrates the proper use of these codes:

```
10 open4,4
20 print#4,"Ahoy! ";
30 print#4,chr$(17);"Ahoy! ";
40 print#4,chr$(14);chr$(17);"Aho
y! ";chr$(145);"AHOY! ";chr$(17);
"Ahoy! ";
50 print#4,chr$(15);chr$(17);"Aho
y! ";
60 print#4,"Ahoy! "
70 close4
```

The listing is in upper/lower case mode to properly display the use of the shifted characters. The CHR$(14) code in line 40 causes double width printing. The printer will continue to print double width until the CHR$(15) is sent in line 50. The program will print the name of our favorite Commodore magazine, seven times, on a single line, while demonstrating the upper/lower case to graphics mode changes.

### *SETTING THE PRINT POSITION*

The CHR$(16) code is similar to the BASIC TAB function. It has two modes of operation. When the printer is in text or character mode, the two bytes which follow immediately after will set the next print position to any of the 80 columns. The actual column number is sent as a two byte string or as the CHR$ codes of the column number. For example:

```
OPEN 4,4
```

can be followed by

```
PRINT#4,CHR$(16)CHR$(52)CHR$(56)"
HELLO"
```

or by

```
PRINT#4,CHR$(16)"48HELLO"
```

Either of these lines will print the word "HELLO" starting with the 48th column. Note that 52 and 56, in the first statement, are the ASCII codes of the digits "4" and "8".

The following program uses the print position code to convert the name of our favorite magazine into a descending staircase motif.

```
10 open4,4,7
20 fori=0to4
30 po$=right$("0"+str$(5*i),2)
40 print#4,chr$(16)po$"Ahoy!"
50 next i
60 print#4,:close4
```

### *SETTING THE DOT POSITION*

The 1525 printer is a dot matrix printer. It is capable of printing individual dots at any point on a page. There are a total of 480 dot positions on an eight inch line. If the CHR$(16) code is preceeded by a CHR$(27) code, then the printhead can be moved to any of these dot positions. As in text mode, the two following bytes determine the dot position. Only this time, it is the actual byte value and not the ASCII code which is used. The first byte is either a 0 or a 1. If it is a 1, 256 is added to the second byte to calculate the position. For example:

```
PRINT#4,CHR$(27)CHR$(16)CHR$(1)CHR
$(15)
```

*Continued on page 85*

# COMMODORE 64

(more power than Apple II at 1/3 the price)

## $188.00*

- 170K Disk Drive $249.00*
- Tractor Friction Printer $169.00*
- 14" Hi-Res Color Monitor $219.00*

*less coupon discount

# COMPUTER AND SOFTWARE SALE

WE HAVE THE BEST SERVICE

WE HAVE THE LOWEST PRICES

# AUTO DIAL MODEM 64

(Best communications package in USA)

## $79.00*

- Computer Learning Pad $49.00
- New Voice Synthesizer $59.00
- Commodore 64 Power for Vic-20 $69.00

### ★ COMMODORE 64 COMPUTER $188.00

You pay only $188.00 when you order the powerful 84K COMMODORE 64 COMPUTER! LESS the value of the SPECIAL SOFTWARE COUPON we pack with your computer that allows you to SAVE OVER $500 off software sale prices!! With only $100 of savings applied, your net computer cost is $88.00!!

### ★ 170 DISK DRIVE $249.00

You pay only $249.00 when you order the 170K Disk Drive! LESS the value of the SPECIAL SOFTWARE COUPON we pack with your disk drive that allows you to SAVE OVER $100 off software sale prices!! With only $500 of savings applied, your net disk drive cost is $149.00

### ★ 80 COLUMN 80CPS TRACTION FRICTION PRINTER $169.00

You pay only $169.00 when you order the Comstar T/F deluxe line printer that prints 8½x11 full size, single sheet, roll or fan fold paper, labels etc. Impact dot matrix, bidirectional. LESS the value of the SPECIAL SOFTWARE COUPON we pack with your printer that allows you to SAVE OVER $500 off software sale prices!! With only $100 of saving applied your net printer cost is only $69.00.

### ★ 14" HI-RES COLOR MONITOR $219.00

You pay only $219.00 when you order this 14" COLOR MONITOR with sharper and clearer resolution than any other color monitors we have tested! LESS value of the SPECIAL DISCOUNT COUPON we pack with your monitor that allows you to save over $500 off software sale prices!! With only $100 of savings applied your net color monitor cost is only $119.00 (16 Colors).

### 80 COLUMN BOARD $99.00

Now you program 80 COLUMNS on the screen at one time! Converts your Commodore 64 to 80 COLUMNS when you plug in the 80 COLUMN EXPANSION BOARD!! PLUS 4 slot expander! Can use with most existing software.

### 80 COLUMNS IN COLOR EXECUTIVE WORD PROCESSOR $49.00

This EXECUTIVE WORD PROCESSOR is the finest available for the COMMODORE 64 computer! The ULTIMATE FOR PROFESSIONAL Word Processing DISPLAYS 40 or 80 COLUMNS IN COLOR or Black and White! Simple to operate, powerful text editing with 250 WORD DICTIONARY, complete cursor and insert/delete key controls line and paragraph insertion, automatic deletion, centering, margin settings and output to all printers! Includes a powerful mail merge.

List $99.00 **SALE $49.00** Coupon $39.00

### SPECIAL SOFTWARE COUPON

We pack a SPECIAL SOFTWARE DISCOUNT COUPON with every COMMODORE 64 COMPUTER DISK DRIVE-PRINTER-MONITOR we sell! This coupon allows you to SAVE OVER $500 OFF SALE PRICES!!

(Examples)
PROFESSIONAL SOFTWARE
COMMODORE 64

| Name | List | Sale | Coupon |
|---|---|---|---|
| Executive Word Processor | $99.00 | $49.00 | $39.00 |
| Executive Data Base | $69.00 | $35.00 | $24.00 |
| 20,000 Word Dictionary | $24.95 | $14.95 | $10.00 |
| Electronic Spread Sheet | $59.95 | $49.00 | $39.00 |
| Accounting Pack | $49.00 | $39.00 | $29.00 |
| Practicalc | $59.95 | $44.95 | $36.95 |
| Programmers Reference Guide | $20.95 | $16.95 | $12.50 |
| Programmers Helper (Disk) | $59.95 | $39.95 | $29.95 |
| 80 Column Screen (Disk) | $59.95 | $39.95 | $29.95 |
| Flip & File Disc Filer | $39.95 | $16.95 | $14.95 |
| Deluxe Tape Cassette | $89.00 | $49.00 | $39.00 |
| Pro Joy Stick | $24.95 | $15.95 | $12.00 |
| Light Pen | $39.95 | $16.95 | $14.95 |
| Dust cover | $8.95 | $6.95 | $4.60 |
| Pogo Joe | $29.95 | $19.95 | $16.95 |
| Pitstop II - Epyx | $39.95 | $29.95 | $26.00* |
| | | | *Plus One FREE |
| Music Calc | $59.95 | $39.95 | $34.95 |
| Filewriter | $59.95 | $39.95 | $34.95 |

*(See over 100 coupon items in our catalog)*
**Write or call for Sample SPECIAL SOFTWARE COUPON!**

### EXECUTIVE QUALITY PROFESSIONAL BUSINESS SOFTWARE

The Cadillac of Business Programs for Commodore 64 Computers

| Item | List | *SALE | Coupon |
|---|---|---|---|
| Inventory Management | $99.00 | $49.00 | $35.00 |
| Accounts Receivable | $99.00 | $49.00 | $35.00 |
| Accounts Payable | $99.00 | $49.00 | $35.00 |
| Payroll | $99.00 | $49.00 | $35.00 |
| General Ledger | $99.00 | $49.00 | $35.00 |

### ★ SUPER AUTO DIAL MODEM $79.00

Easy to use. Just plug into your Commodore 64 computer and you're ready to transmit and receive messages. Easier to use than dialing your telephone just push one key on your computer! Includes exclusive easy to use program for up and down loading to printer and disk drives. List $129.00 **SALE $79.00.**

### NEW COMPUTER LEARNING PAD $49.00

makes other graphics tablet obsolete. This new TECH SCETCH LEARNING PAD allows you to draw on your T.V. or Monitor and then you can print whatever you draw on the screen on your printers. FANTASTIC!!! List $79.95 **SALE $49.00**. Coupon $39.95

### NEW VOICE SYNTHESIZER $59.00

For Com-64 or VIC-20 computers. Just plug it in and your can program words and sentences, adjust volume and pitch, make talking adventure games, sound action games and customized talkies!! FOR ONLY $19.95 you can add TEXT TO SPEECH, just type a word and hear your computer talk—ADD SOUND TO "ZORK", SCOTT ADAMS AND AARDVARK ADVENTURE GAMES!! (Disk or tape).

### COM-64 POWER FOR VIC-20 $69.00

Just plug in our 32K RAM MEMORY EXPANDER and you get as much usable programming power as the Commodore-64 computer!! Master control switches on cover. Gold Edge connectors, five year warranty (FREE $29.95 CARTRIDGE GAME).

### 16K RAM CARTRIDGE $49.00

Increases VIC-20 programming power 4 times. Expands total memory to 41K (41,000 bytes). Memory block switches are on outside cover! CARDCO Includes FREE $29.95 game!!

### 9" GREEN SCREEN MONITOR $69.95

Excellent quality SANYO, easy to read, 80 columns x 24 lines. Green Phosphorous screen with anti-glare, metal cabinet! Saves your T.V. PLUS $9.95 for connecting cable. Com-64 or VIC-20.

### 12" GREEN OR AMBER MONITOR $99.00

Your choice of green or amber screen monitor top quality, SANYO 80 columns x 24 lines, easy to read, anti-glare, faster scanning! PLUS $9.95 for connecting cable. Com-64 or VIC-20.

**PHONE ORDERS**
**8AM - 8PM Weekdays**
**9AM - 12N Saturdays**

• LOWEST PRICES • 15 DAY FREE TRIAL • 90 DAY FREE REPLACEMENT WARRANTY
• BEST SERVICE IN U.S.A. • ONE DAY EXPRESS MAIL • OVER 500 PROGRAMS • FREE CATALOGS

Add $10.00 for shipping, handling and insurance. Illinois residents please add 6% tax. Add $20.00 for CANADA, PUERTO RICO, HAWAII, ALASKA, APO-FPO orders. Canadian orders must be in U.S. dollars. WE DO NOT EXPORT TO OTHER COUNTRIES.

Enclose Cashiers Check, Money Order or Personal Check. Allow 14 days for delivery, 2 to 7 days for phone orders, 1 day express mail!

VISA — MASTER CARD — C.O.D.

**PROTECTO ENTERPRIZES** (WE LOVE OUR CUSTOMERS)

**BOX 550, BARRINGTON, ILLINOIS 60010**
**Phone 312/382-5244 to order**

Reader Service No. 20

NEW 128K —MEGA BYTE DUAL DISK DRIVE—80 COLUMN

# COMPUTER SYSTEM SALE!

## HOME • BUSINESS • WORD PROCESSING




*LOOK AT ALL YOU GET FOR ONLY* **$895.**

| | LIST PRICE |
|---|---|
| ① B128 COMMODORE 128K 80 COLUMN COMPUTER | $ 995.00 |
| ② 4023 - 100 CPS - 80 COLUMN BIDIRECTIONAL PRINTER | 499.00 |
| ③ 8050 DUAL DISK DRIVE (over 1 million bytes) | 1795.00 |
| ④ 12" HI RESOLUTION 80 COLUMN MONITOR | 249.00 |
| • BOX OF 10 LORAN LIFETIME GUARANTEED DISKS | 49.95 |
| • 1100 SHEETS FANFOLD PAPER | 19.95 |
| • ALL CABLES NEEDED FOR INTERFACING | 102.05 |
| TOTAL LIST PRICE | **$3717.95** |

**$895** SALE PRICE

### PLUS YOU CAN ORDER THESE BUSINESS PROGRAMS AT SALE PRICES

| | LIST | SALE |
|---|---|---|
| Professional 80 Column Word Processor | $149.95 | **$99.00** |
| Professional Data Base | $149.95 | **$99.00** |
| Accounts Receivable | $149.95 | **$99.00** |
| Accounts Payable | $149.95 | **$99.00** |
| Payroll | $149.95 | **$99.00** |
| Inventory | $149.95 | **$99.00** |
| General Ledger | $149.95 | **$99.00** |
| Financial Spread Sheet | $149.95 | **$99.00** |

### PRINTER REPLACEMENT OPTIONS

(replace the 4023 with the following at these sale prices)

| | LIST | SALE |
|---|---|---|
| Olympia Executive Letter Quality Serial Printer | $699.00 | **$399.00** |
| Comstar Hi-Speed 160 CPS 15½" Serial Business Printer | $779.00 | **$499.00** |
| Telecommunications Deluxe Modem Package | $199.00 | **$139.00** |

**15 DAY FREE TRIAL**. We give you 15 days to try out this SUPER SYSTEM PACKAGE!! If it doesn't meet your expectations, just send it back to us prepaid and we will refund your purchase price!!
**90 DAY IMMEDIATE REPLACEMENT WARRANTY**. If any of the SUPER SYSTEM PACKAGE equipment or programs fail due to faulty workmanship or material we will replace it IMMEDIATELY at no charge!!

**Add $50.00 for shipping and handling!!**
**$100.00 for Alaska and Hawaii orders.**
WE DO NOT EXPORT TO OTHER COUNTRIES
Enclose Cashiers Check, Money Order or Personal Check. Allow 14 days for delivery, 2 to 7 days for phone orders, 1 day express mail! We accept Visa and MasterCard. We ship C.O.D. to continental U.S. addresses only.

**PROTECTO**
**ENTERPRIZES** (WE LOVE OUR CUSTOMERS)
**BOX 550, BARRINGTON, ILLINOIS 60010**
**Phone 312/382-5244 to order**

# 14" COLOR MONITOR SALE!!!

(Lowest price in USA)




14" Color Computer Monitor

16 COLORS

- Built in speaker and audio
- Front Panel Controls
- For Video Recorders
- For Small Business/ Computers
- Apple-Commodore Atari-Franklin-etc.

- Beautiful Color Contrast
- High Resolution
- Sharp Clear Text
- 40 Columns x 24 lines
- List $399

**SALE $219**

*15 Day Free Trial - 90 Day Immediate Replacement Warranty*

**12" ZENITH® HI-RESOLUTION GREEN OR AMBER TEXT DISPLAY MONITOR**
List $249 **SALE$119**

80 Columns x 24 lines, Hi-Resolution-crisp clear easy to read text with anti glare screen! A Must for word processing.

**12" SANYO GREEN OR AMBER SCREEN MONITOR** List $199 **SALE $99**

80 Columns x 24 lines, amber or green text display, easy to read, no eye strain, up front controls.

**9" SANYO GREEN SCREEN DATA MONITOR** List $149 **SALE $69**

80 Columns x 24 lines easy to read, up front controls metal cabinet.

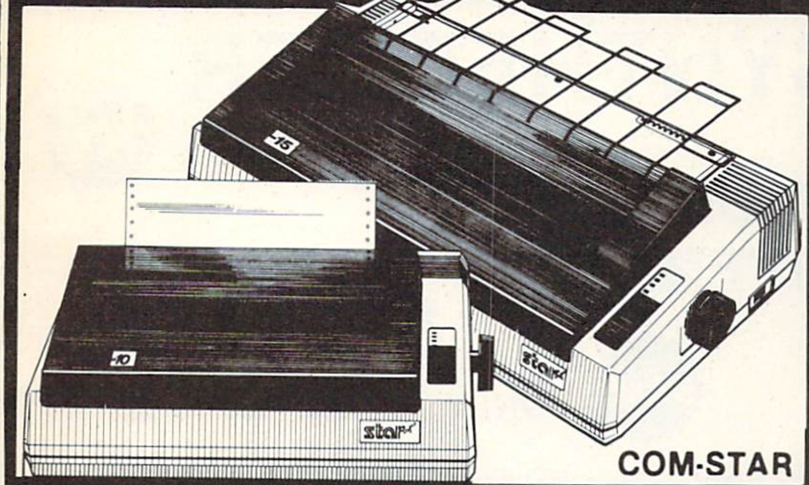
• LOWEST PRICES • 15 DAY FREE TRIAL • 90 DAY FREE REPLACEMENT WARRANTY
• BEST SERVICE IN U.S.A. • ONE DAY EXPRESS MAIL • OVER 500 PROGRAMS • FREE CATALOGS

Add $10.00 for shipping, handling and insurance. Illinois residents please add 6% tax. Add $20.00 for CANADA, PUERTO RICO, HAWAII orders. WE DO NOT EXPORT TO OTHER COUNTRIES.

Enclose Cashiers Check, Money Order or Personal Check. Allow 14 days for delivery, 2 to 7 days for phone orders, 1 day express mail! Canada orders must be in U.S. dollars. Visa - MasterCard - C.O.D.

**PROTECTO**
**ENTERPRIZES** (WE LOVE OUR CUSTOMERS)
**BOX 550, BARRINGTON, ILLINOIS 60010**
**Phone 312/382-5244 to order**

# FANTASTIC COMPUTER PRINTER SALE!!!




## COM-STAR T/F

Tractor
Friction
Printer

only $ 169**

- Lowest Priced, Best Quality, Tractor-Friction Printers in the U.S.A.
- Fast 80-120-160 Characters Per Second
- 40, 46, 66, 80, 96, 132 Characters Per Line Spacing
- Word Processing
- Print Labels, Letters, Graphs and Tables
- List Your Programs
- Print Out Data from Modem Services
- "The Most Important Accessory for Your Computer"

### ** DELUXE COMSTAR T/F 80 CPS Printer — $169.00

This COMSTAR T/F (Tractor Friction) PRINTER is exceptionally versatile. It prints 8½" x 11" standard size single sheet stationary or continuous feed computer paper. Bi-directional, impact dot matrix, 80 CPS, 224 characters. (Centronics Parellel Interface).

### Premium Quality 120-140 CPS 10" COM-STAR PLUS+ Printer $249.00

The COM-STAR PLUS+ gives you all the features of the COMSTAR T/F PRINTER plus a 10" carriage, 120-140 CPS, 9 x 9 dot matrix with double strike capability for 18 x 18 dot matrix (near letter quality), high resolution bit image (120 x 144 dot matrix), underlining, back spacing, left and right margin settings, true lower decenders with super and subscripts, prints standard, italic, block graphics and special characters. It gives you print quality and features found on printers costing twice as much!! (Centronics Parallel Interface) (Better than Epson FX80). List $499.00 **SALE $249.00**

### Premium Quality 120-140 CPS 15½" COM-STAR PLUS+ Business Printer $349.00

Has all the features of the 10" COM-STAR PLUS+ PRINTER plus 15½" carriage and more powerful electronics components to handle large ledger business forms! (Better than Epson FX 100). List $599

**SALE $349.00**

### Superior Quality 10" COM-STAR+ H.S. HIGH SPEED 160-180 CPS Business Printer $369.00

This **Super High Speed Com-Star+ Business Printer** has all the features of the 10" COM-STAR+ PRINTER with HIGH SPEED BUSINESS PRINTING 160-180 CPS, 100% duty cycle, 8K Buffer, diverse character fonts, special symbols and true decenders, vertical and horizontal tabs. **A RED HOT BUSINESS PRINTER** at an unbelievable low price (Serial or Centronics Parallel Interface) List $699.00 **Sale $369.00.**

### Superior Quality 15½" COM-STAR PLUS+ H.S. High Speed 160 - 180 CPS Business Printer $469.00

This **Super High Speed COM-STAR+ 15½" Business Printer** has all the features of the 10" COM-STAR BUSINESS PRINTER witha 15½" Carriage and more powerful electronic components to handle larger ledger business forms! Exclusive bottom feed. (Serial Centronics Parallel Interface) List $799.00 **Sale $469.00**

### Olympia

### Executive Letter Quality DAISY WHEEL PRINTER $379.00

This is the worlds finest daisy wheel printer **Fantastic Letter Quality**, up to 20 CPS bidirectional, will handle 14.4" forms width! Has a 256 character print buffer, special print enhancements, built in tractor-feed (Centronics Parallel and RS232C Interface) List $699 **SALE $379.**

## • *15 Day Free Trial - 1 Year Immediate Replacement Warranty*

### PARALLEL INTERFACES

For VIC-20 and COM-64 — $49.00 For Apple computers — $79.00 Atari 850 Interface — $79.00 For ALL IBM Computers — $89.00

**Add $14.50 for shipping, handling and insurance. Illinois residents please add 6% tax. Add $29.00 for CANADA, PUERTO RICO, HAWAII, ALASKA, APO-FPO orders. Canadian orders must be in U.S. dollars. WE DO NOT EXPORT TO OTHER COUNTRIES.**

**Enclose Cashiers Check, Money Order or Personal Check. Allow 14 days for delivery, 2 to 7 days for phone orders, 1 day express mail!**

VISA—MASTER CARD—We Ship C.O.D. to U.S. Addresses Only

## PROTECTO ENTERPRIZES (WE LOVE OUR CUSTOMERS)

**BOX 550, BARRINGTON, ILLINOIS 60010**
**Phone 312/382-5244 to order**

COM-STAR PLUS+ ABCDEFGHIJKLMNOPQRSTUVWXYZ
Print Example: ABCDEFGHIJKLMNOPQRSTUVWXYZ 1234567890

## COMMODORE-64 or VIC-20

# VOICE SYNTHESIZER

NEW

NEW

## MAKE YOUR COMPUTER TALK

VOTRAX BASED HARDWARE

SALE

ONLY **$59.00**

You can program any words or sentences • Adjust volume and pitch • Make adventure games that talk • Real sound action games • Make customized talkies • (Demo disk or tape included) • Requires Speaker

You can add **TEXT TO SPEECH SOFTWARE** that allows you to simply type what you want to hear!! Also allows you to add sound and voice to **SCOTT ADAMS AARDVARK** and **"ZORK" ADVENTURE GAMES** List $29.95 Sale $19.95 (Disk or Tape).

• LOWEST PRICES • 15 DAY FREE TRIAL • 90 DAY FREE REPLACEMENT WARRANTY
• BEST SERVICE IN U.S.A. • ONE DAY EXPRESS MAIL • OVER 500 PROGRAMS • FREE CATALOGS

## COMMODORE 64

# 80 COLUMN BOARD

SALE

FOR ONLY **$99.00**




Now you can program 80 columns on the screen at one time! Converts your Commodore 64 to 80 columns when you plug in the PROTECTO 80 Expansion Board. List $199.00. **Sale $99.00.**

**Includes 4 Slot Expander and can be used with most existing software!!!**

Add $3.00 for shipping, handling and insurance. Illinois residents please add 6% tax Add $6.00 for CANADA, PUERTO RICO, HAWAII, ALASKA, APO-FPO orders. Canadian orders must be in U.S. dollars. WE DO NOT EXPORT TO OTHER COUNTRIES.
Enclose Cashiers Check, Money Order or Personal Check. Allow 14 days for delivery, 2 to 7 days for phone orders, 1 day express mail!
VISA — MASTER CARD — C.O.D.
No C.O.D. to Canada, APO-FPO.

*WE LOVE OUR CUSTOMERS*

**PROTECTO**
**ENTERPRIZES** (WE LOVE OUR CUSTOMERS)
**BOX 550, BARRINGTON, ILLINOIS 60010**
**Phone 312/382-5244 to order**

COMMODORE 64

# Computer Learning Pad

**Coupon 39^95^**

List $79.95

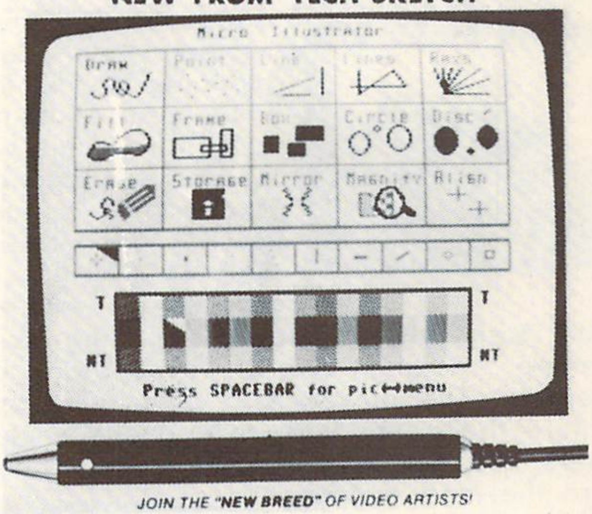
**Sale $49.95**

- **Makes graphic tablets Obsolete**
- **Includes Special Printer Feature.**

Now you get this **Fantastic Tech Sketch Computer Learning Pad Program** with a **Professional Light Pen** Free! (39.95 value) plus the **Micro Illustrated Graphics Pad Program** that allows you to draw on your T.V. or monitor screen (better than Gibsons $99 light pen). Whatever you can draw on the screen you can print on your printer (a Tech Sketch Exclusive.)

## Hi-Resolution!

**NEW FROM TECH SKETCH**




*JOIN THE "NEW BREED" OF VIDEO ARTISTS!*

## Commodore 64 & VIC-20

**$24.95 Professional Model $24.95**

# Light Pens

**Professional Light Pen.** This is the TECH SKETCH Hi-Resolution model that allows you to draw fantastic pictures and professional diagrams — single-pixel accuracy! (Includes famous paint and sketch program.) List $39.95. **Sale $24.95.** *Coupon $22.95*

**$24.95**

**Economy Model with 3 programs $16.95**

Add $3.00 for shipping, handling and insurance. Illinois residents please add 6% tax. Add $6.00 for CANADA, PUERTO RICO, HAWAII, ALASKA, APO-FPO orders. Canadian orders must be in U.S. dollars. WE DO NOT EXPORT TO OTHER COUNTRIES.
Enclose Cashiers Check, Money Order or Personal Check. Allow 14 days for delivery, 2 to 7 days for phone orders, 1 day express mail!
VISA — MASTER CARD — C.O.D.
No C.O.D. to Canada, APO-FPO.

**PROTECTO**

**ENTERPRIZES** (WE LOVE OUR CUSTOMERS)

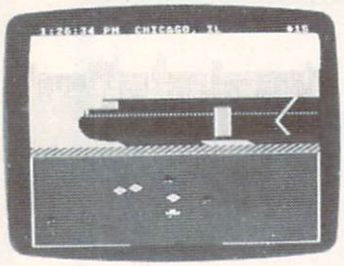
**BOX 550, BARRINGTON, ILLINOIS 60010**
**Phone 312/382-5244 to order**

# Scholastic Software

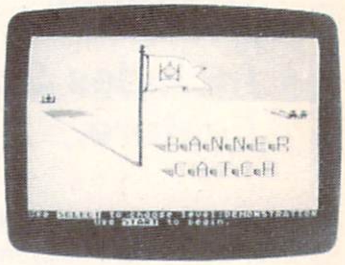
## Commodore 64

# Super School Savings

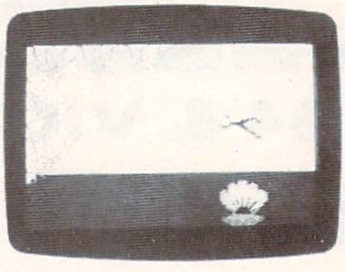
**0800 Agent U.S.A.** Agent USA challenges the player to intercept the Fuzzbomb by travelling across the U.S.A. You must use your knowledge of time zones, state capitals, geography, map reading, and route planning to achieve your goal. Fantastic adventure game with full color graphics and excellent sound. (Disk.) List $34.95. **Sale $24.95.**



**0804 Banner Catch.** This one even teaches Binary numbers. Players must work together as a team in this capture the flag game where you must get Max's flag before his robot raiders get yours. Fantastic play action and teamwork and cooperation skills make this a must for everyone. (Disk.) List $34.95. **Sale $24.95.**



**0808 Spell Diver.** Sharpen spelling skills, improve reading, word recognition and retention. The player is a diver who must explore the ocean floor to find and uncover letters. The more quickly a player can guess a word the more points are scored. Sea creatures and other dangerous surprises plus a time limit make it more exciting. Fantastic graphics and sound. (Disk.) List $34.95. **Sale $24.95.**



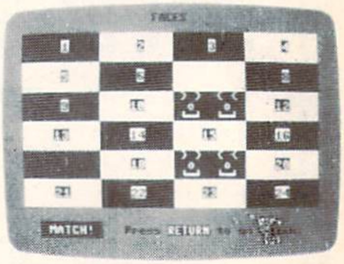
**0812 Double Feature Adventure.** Two full color graphic adventures that allow the player to choose among danger, humor, excitement and chance. Plus these programs make the players stars of the adventure. Explore the microzone or the Northwoods while improving reading skills and building vocabulary. (Disk.) List $29.95. **Sale $17.95.**




**0816 Tales Of Mystery.** Two more famous twist-a-plot adventures that challenge the player first, to find a friend lost in a haunted house. Second, to find a statue stolen from an island. Fantastic graphic adventures. (Disk.) List $29.95. **Sale $17.95.**




**0820 Square Pairs.** Sharpen and improve your vocabulary, reading, and spelling skills while improving deduction and memory. You must match characters in the ready to play games then you can create your own. (1 to 4 players.) (Disk.) List $29.95. **Sale $17.95.**




Add $3.00 for shipping, handling and insurance. Illinois residents please add 6% tax. Add $6.00 for CANADA, PUERTO RICO, HAWAII, ALASKA, APO-FPO orders. Canadian orders must be in U.S. dollars. WE DO NOT EXPORT TO OTHER COUNTRIES.

Enclose Cashiers Check, Money Order or Personal Check. Allow 14 days for delivery, 2 to 7 days for phone orders, 1 day express mail!

VISA — MASTER CARD — C.O.D.

No C.O.D. to Canada, APO-FPO.

**PROTECTO**

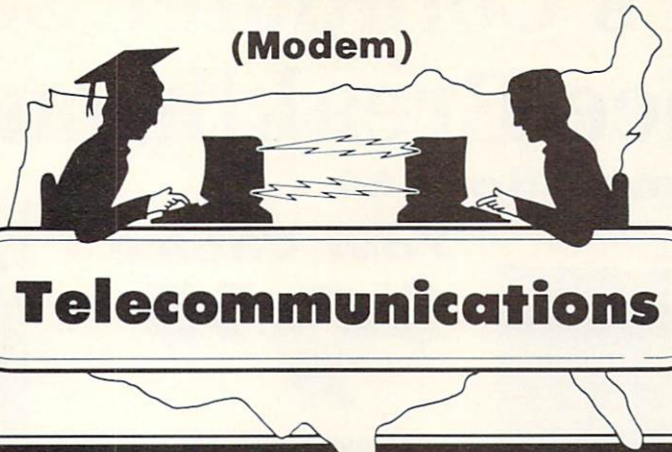
**ENTERPRIZES** (WE LOVE OUR CUSTOMERS)

BOX 550, BARRINGTON, ILLINOIS 60010
Phone 312/382-5244 to order

# Commodore 64

Sale

Sale




$79

$79

## FOR CHILDREN·ADULTS·BUSINESS

# Complete Auto Dial Telecommunications Package

*"The only telecommunications package you will ever need."*

NEW

NEW

## *Total Telecommunications*

**• 300 Baud Modem • Auto Dial • Auto Answer • Upload & Download**

*• Membership in 52 Database Services (UPI News)*

## Reach Out and Access Someone

- Educational courses
- Financial Information
- Banking at Home
- Popular Games
- News Updates and Information
- Electronic Shopping
- Research and Reference Materials

**The Total Telecommonications Package offers you all this plus ...**

- Auto Log-on
- Dialing from Keyboard
- On-line Clock
- Capture and Display High Resolution Characters
- Download Text, Program or Data Files
- Stores on Disk Downloaded Files
- Reads Files from Disk and Uploads Text or Program Files
- Select Any Protocol (access almost any computer or modem)
- Plus Much, Much More

List $129.95 **Special Low Christmas Price** **$79.00**

**We are so sure this is the only telecommunications package you will need we will give you 15 days Free Trial.**

Add $3.00 for shipping, handling and insurance. Illinois residents please add 6% tax. Add $6.00 for CANADA, PUERTO RICO, HAWAII, ALASKA, APO-FPO orders. Canadian orders must be in U.S. dollars. WE DO NOT EXPORT TO OTHER COUNTRIES.

Enclose Cashiers Check, Money Order or Personal Check. Allow 14 days for delivery, 2 to 7 days for phone orders, 1 day express mail!

VISA — MASTER CARD — C.O.D.

No C.O.D. to Canada, APO-FPO.

**PROTECTO**
**ENTERPRIZES** (WE LOVE OUR CUSTOMERS)
**BOX 550, BARRINGTON, ILLINOIS 60010**
**Phone 312/382-5244 to order**

Reader Service No. 27

# Hayden's Computer Software Price Breakthrough

## Commodore 64

***Your Choice***

**$9.95**

Originally $24.95 to $34.95













### Text Adventure

**Crime Stopper:**
Assume the role of a private eye to save a kidnapped heiress.

**Crystal Caverns:**
Find hidden buried treasures under an old mansion.

### Strategy Games

**Final Conflict:**
Program robots to destroy enemy's base. Battlefield variations game can be played against computer or opponent.

**King Cribbage:**
Graphic version of the classic card game. Hi-res graphic representation of playing cards and cribbage board.

**Reversal:**
Winner of the national "Othello" tournament.

**Championship Golf:**
Select club, strike, direction; a must for every golfer.

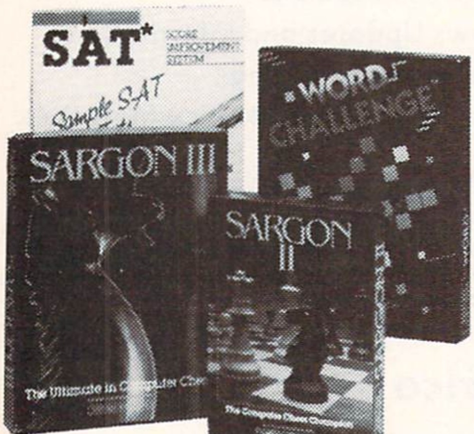
### Education

**MicroAddition**
**MicroSubtraction**
**MicroMultiplication**
**MicroDivision**
Color animation and cheerful tunes make learning arithmetic fun.

**Match-Up:**
3 matching games for pre-school that use colorful objects that teach the principles of "same," "related," and "different."

**Don't Shoot That Word:**
Your beginning readers will become skilled at word recognition with this fun game.

## Other Great Values From Hayden




### Sargon II:
The best selling microcomputer chess game in the world. A classic game with seven levels of play.

was $34.95
now **$17.95**

### Word Challenge:
An engaging word game like Boggle, 90,000 word lexicon, dozens of game variations.

was $39.95
now **$29.95**

### SAT Sample Tests:
It includes a diagnostic Pre-test for determining strengths and weaknesses in the mathematical and verbal section of the SAT. Offers test-taking strategies and two timed simulated SAT exams scored on the "800" scale.

was $34.95
now **$17.95**

Add $3.00 for shipping, handling and insurance. Illinois residents please add 6% tax. Add $6.00 for CANADA, PUERTO RICO, HAWAII, ALASKA, APO-FPO orders. Canadian orders must be in U.S. dollars. WE DO NOT EXPORT TO OTHER COUNTRIES.
Enclose Cashiers Check, Money Order or Personal Check. Allow 14 days for delivery, 2 to 7 days for phone orders, 1 day express mail!
VISA — MASTER CARD — C.O.D.
No C.O.D. to Canada, APO-FPO.

**PROTECTO**
**ENTERPRIZES** (WE LOVE OUR CUSTOMERS)
**BOX 550, BARRINGTON, ILLINOIS 60010**
**Phone 312/382-5244 to order**

Reader Service No. 28

# Avalon Hill

## World Renowned Strategy Games of the Masters

### ☆ ☆ ☆ *Super Strategy Sale* ☆ ☆ ☆

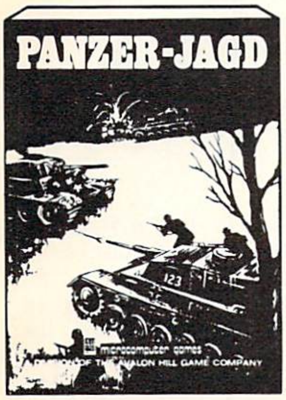



**Telengard-Real Time**. This is the game all the dungeon fans have been waiting for. Use wits, magic, and fast thinking to overcome the 50 levels of the TELENGARD. Fantastic graphics and action. (Disk.) List $29.95. **Sale $19.95**. (Tape.) List $24.95. **Sale $16.95**.

**T.A.C. (Tactical Armor Command.)** Fantastic computer adaptation of the famous board game. The player or players control up to 8 units simultaneously including tanks, infantry and anti-tank guns. (Fantastic graphics.) (Disk.) List $44.95. **Sale $34.95**.




# Commodore 64

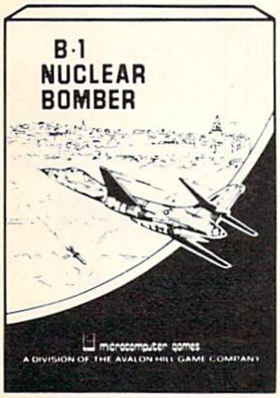



**Panzer Jagd**. Fantastic tank simulation puts you in the German seat forced to secure the area of the Russian front. If you succeed your remaining units are saved for the big battle. See if you can rewrite history or make the same mistakes. (Great playability.) (Disk.) List $29.95. **Sale $21.95**.

**Legionnaire**. You as Caesar Augustus command up to 10 legions against an army of barbarian infantry, all aspects including cavalry, shock effects, fatigue, morale and slope effects as the battlefield scrolls before you in full color and animation. (Tape.) List $39.95. **Sale $24.95**. (Disk.) List $44.95. **Sale $27.95**.

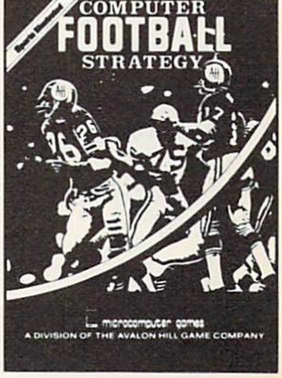



# Commodore 64




**B-1 Nuclear Bomber**. Pilot a B-1 bomber through stiff Russian defenses on the way to the target city. The computer controls Soviet MIG fighters and surface to air missiles. You must rely on your electronic counter measures and self defense missiles to succeed. Fantastic strategy game. (Tape.) List $19.95. **Sale $13.95**. (Disk.) List $24.95. **Sale $18.95**.

**Computer Football**. Match wits against the computer or another live opponent. This fantastic game based on the award winning Sports Illustrated game of professional football will keep you going for hours. A must for all the Armchair Quarterbacks of the world. (Tape). List $19.95. **Sale $13.95**. (Disk.) List $24.95. **Sale $18.95**.




Add $3.00 for shipping, handling and insurance. Illinois residents please add 6% tax. Add $6.00 for CANADA, PUERTO RICO, HAWAII, ALASKA, APO-FPO orders. Canadian orders must be in U.S. dollars. WE DO NOT EXPORT TO OTHER COUNTRIES.

Enclose Cashiers Check, Money Order or Personal Check. Allow 14 days for delivery, 2 to 7 days for phone orders, 1 day express mail!

VISA — MASTER CARD — C.O.D.

No C.O.D. to Canada, APO-FPO.

# PROTECTO

**ENTERPRIZES** (WE LOVE OUR CUSTOMERS)

**BOX 550, BARRINGTON, ILLINOIS 60010**
**Phone 312/382-5244 to order**

EPYX
COMPUTER SOFTWARE
Strategy Games for the Action-Game Player

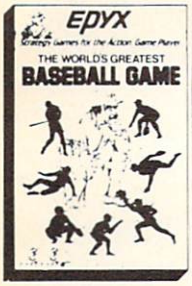
# * Buy 1 (at Sale Price!) Get One FREE

## Commodore 64

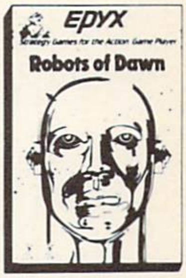
(Expires 12-25-84)

* (Limit: One Per Customer)

# The Best Arcade Games Around

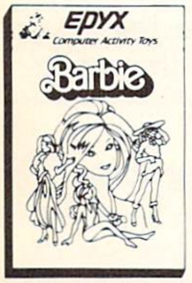



**World's Greatest Baseball** — Finally a great sports game. Realistic right down to the playing field. Contro.l all the actions -even the outfielders. Play realistic baseball all year round. (Fantastic graphics, action, and sound!) (Disk) List $34.95. **Sale $27.95.**




**Robots of Dawn** — Elijah Baley is here from the best-selling science fiction series by world renowned author Isaac Asimov. Now you can become Earth's most famous detective in this exciting test-adventure. Question the inhabitants of far flung cultures — who's lying, who's telling the truth, and who is trying to murder you? Jump into the action in this fascinating epic murder mystery. Can you piece together the elusive clues to solve the eternal question "who done it?" (Disk) List $39.95. **Sale $27.95.**

**PROTECTO EXCLUSIVE**




**Barbie** — The biggest name in dolls. Browse through Barbie's closet full of beautiful clothing and dress her for the party. You can cut her hair or make it longer and color it or change the style. Buy new clothing at the boutique or any of the six other specialty shops, or even go to the dress shop and create Barbie's designer clothes with the help of the computer. The combinations are endless and so is the fun. One or two players. (Disk) List $39.95. **Sale $27.95.**




**G.I. Joe** — The best selling toy soldier is now available as an activity toy on the home computer. Select the battle situation then choose the equipment you think you will need to get the job done. Choose from a number of uniforms and weapons in your well stocked arsenal and get ready for the action. Play alone or with a friend, if you plan the right strategy you will complete the mission if not you will have to try again. One or two players. (Disk) List $39.95. **Sale $27.95.**

**PROTECTO EXCLUSIVE**




**Hot Wheels** — Now all the action and fun from playing with Hot Wheels cars is taken one step further on the home computer. Hot Wheels lets you choose your play activity from repairing cars to the destruction derby. You can actually play the activities along with a partner on the computer screen, even build and customize your own cars. A whole new way to play Hot Wheels. One or two players. (Disk) List $39.95. **Sale $27.95.**




**Break Dance** — Break Dance is an action game in which your dancer tries to break through a gang of break dancers descending on him. "A simon-like game" where your dancer has to keep the steps of the computer controlled dancer, and even a free dance segment where you develop your own dance routines and the computer plays them back for you to watch. Now anyone can break dance!! (Disk) List $39.95 **Sale $27.95.**

* Buy Epyx games from Protecto and choose Silicon Warrior, Lunar Outpost, or Upper Reaches of Apshai FREE! ($39.95 value)

In case of stock outage we will substitute.

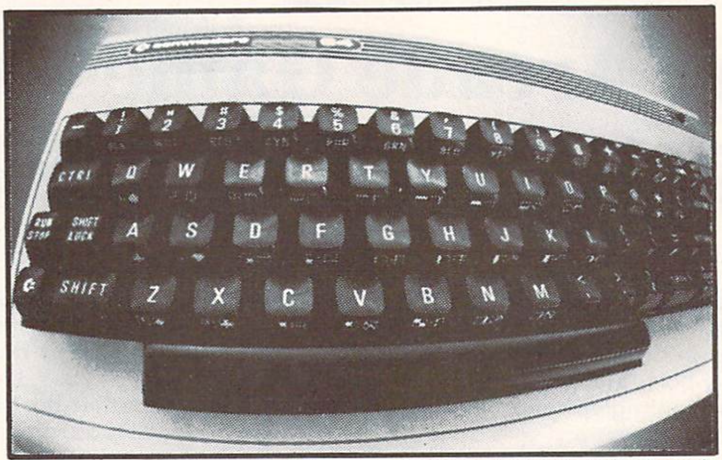
Add $3.00 for shipping, handling and insurance. Illinois residents please add 6% tax. Add $6.00 for CANADA, PUERTO RICO, HAWAII, ALASKA, APO-FPO orders. Canadian orders must be in U.S. dollars. WE DO NOT EXPORT TO OTHER COUNTRIES.
Enclose Cashiers Check, Money Order or Personal Check. Allow 14 days for delivery, 2 to 7 days for phone orders, 1 day express mail!
VISA — MASTER CARD — C.O.D.
No C.O.D. to Canada, APO-FPO.

**PROTECTO**
**ENTERPRIZES** (WE LOVE OUR CUSTOMERS)
**BOX 550, BARRINGTON, ILLINOIS 60010**
**Phone 312/382-5244 to order**

Reader Service No. 31

For The C64

# "Introducing the Word Processor with Less."



WORD COMMANDER 64: Your Commodore never again at a loss for words.

You have to look very closely to find what we left out of our new word processor. But once you try the WORD COMMANDER 64 you'll discover what's missing. We've taken the *process* out of word processing and put it where it belongs—in our software. With the WORD COMMANDER 64 there's less for you to do and more that it does. And that makes *you* look good.

## Only Two Hands.

The WORD COMMANDER 64 is a complete word processor. We've included every function you'll need and some comforts you'll enjoy. And we didn't hide them in complicated codes or commands that require three hands. Everything about the WORD COMMANDER 64 makes plain sense.

## Less Hassle.

When you write you probably change your mind a lot. Now changing your text is almost as easy as changing your mind. Like when you want to underline a word or group of words all you do is press "COMMODORE-U" —regardless of the printer control code your printer uses. Simple.

## Less is More.

While simple to use, you won't find the WORD COMMANDER 64 lacking in power or sophistication. We've integrated all of the best features into the WORD COMMANDER 64:

- extremely user friendly
- menu driven
- full word wrap
- block text right
- center text
- right justify text
- delete a block of text
- insert a block of text
- move a block of text
- copy a block of text
- three different fonts supported
- print in bold type
- print in elongated type
- allows sending of any control codes to your printer
- select any background color of your preference
- select any character color as well
- select cursor color
- set top, right, bottom and left margin
- set line spacing
- print headers and footers on every page
- automatically number pages
- print subscripts and superscripts
- underline text
- search for specific text
- search and replace
- change margins, font, or justification from within your document
- full 80 column view of the page before printing
- use MMG's MAIL LIST for mail merge capability

## Follow Your Ideas.

The words you need to write represent your ideas; and they flow—one after another. So we designed the WORD COMMANDER 64 to work along with your thought processes. And not slow you down. We did that by making it less complicated and easier to use. So easy you probably already know how to use it.

Oh, there's one more thing missing from the WORD COMMANDER 64, the high price: Only $49.95

Includes full manual and tutorial.

## How to Get Yours.

Available at your local dealer or direct from **MMG Micro Software**. Just send check or money order to:

**P.O. Box 131 Marlboro, NJ 07746**

Or for MasterCard, Visa, and C.O.D. deliveries call: **(201) 431-3472**

Please add $3.00 for postage and handling for all direct orders. New Jersey residents add 6% sales tax.

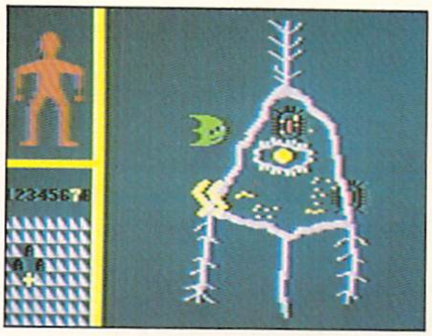
MMG micro software

## *CELL DEFENSE*
***HesWare***
***Commodore 64***
***Disk; $29.95***

Few real-life situations adapt themselves as neatly to the arcade game format as the life-or-death struggle that occurs in the human bloodstream between the body's immune system and viral invaders. So closely does this microdrama parallel many a popular videogame that children who play *Cell Defense* should find it no less entertaining to zap viruses than spaceships or killer robots.

You want villains? You got 'em, in the form of viruses that attack healthy cells (another videogame staple: helpless victims) and commandeer their chemical supplies to produce more viruses and spread disease throughout the body. You want heroes? They come under a variety of names: interferon, macrophages, antibodies, and committed T-cells. Like good game characters should, these various cell defenses have abilities as diverse as their names. The game's designers took no scientific liberties in so endowing them: nature was imaginative enough. Interferon cannot help infected cells, but can stop viruses from reproducing inside healthy ones. Macrophages (meaning "big eaters") gobble up viruses that have not yet entered cells. Antibodies patrol the bloodstream and destroy viruses that enter their area. Committed T-Cells attach themselves to infected cells and blow them up.

The playfield is even formatted in a manner familiar to game fans: multiple battlefronts (cells), one onscreen at a time, which the player traverses by left-right and up-down scrolling while watching for attacks via an onscreen scanner. You carry defenses from cell to cell with your cursor, toting



***Your object is to catch a virus.***
***READER SERVICE NO. 46***

committed T-cells to cells that are hopelessly overrun, macrophages to areas swarming with loose viruses, etc.

The illustration shows a typical cell, with the scanner in the lower left hand corner, and a representation of the patient above it. The figure turns green, thermometer-style, to indicate how sick the patient is. If the viruses destroy half your patient's cells, 60% of his labile cells, 50% of his stable (slowly reproducing) cells, or 40% of his perennial (non-reproducing) cells, he turns as green as a frog—and croaks. To win, you must wipe out all the viruses before the damage goes too far. Difficulty level is easily controlled by adjusting four variables: number of cell layers, virus reproduction rate, cell regeneration rate, and health of organism.

The graphics, in addition to being colorful and engaging, are accurate. A child should have no trouble recognizing a stable cell after seeing it repeatedly in this game. Whether or not he'll ever be required to recognize a stable cell is another question. *Cell Defense* teaches and reinforces the meanings of about a dozen biological terms. The two high school bio teachers we spoke to felt it was unlikely that the body's immune system would be covered in sufficient depth in a high school class to include even half of the dozen terms. And even if it were, your child is still learning only a dozen terms. At $29.95, that's over $2.00 a term. It might not have been possible to make the game broader-based without sacrificing the drama of the gameplay, but...chee, $2.00 a term.

The manual (designed to resemble a notebook) warns children that *Cell Defense* offers only an ultra-simplified introduction to the subject of the body's immune sys-




**Trivia Fever** ***(see page 36) includes multiple handicapping facilities.***
***READER SERVICE NO. 47***

tem, and recommends several readings. If the game can stimulate further investigation, it's worth the price. And this game is well-constructed enough to do exactly that for many children.

HesWare, 150 North Hill Drive, Brisbane, CA 94005 (phone: 415-468-4111).

*–Greg Fried*

### *TRIVIA FEVER*
***Professional Software, Inc.***
***Commodore 64***
***Disk; $39.95***

In reviewing *Trivia Fever,* the comparison to *Trivial Pursuit* is unavoidable. If trivia grabs you, you probably already own the latter game, or have access to it through a friend. Why do you need a trivia game that you can play on your 64?

You don't. If you've already got *Trivial Pursuit,* we'd pass this one up. Trivia is trivia is trivia. But if you're looking to make your first trivia game purchase, *Trivia Fever* has several advantages over its famous precursor.

The gameplay offers no surprises – it's just trivia questions, lots of them, in seven categories (from which each player selects five): science and technology, geography, history, sports, films and entertainment, famous people, and nature and animals. To win, you or your team must answer the user-designated number of questions in each of five categories, then a category completion question for each, and finally a game-winning question.

No big differences there. *Trivia Fever* is different in its use of the computer screen. No graphics are offered, and very little effort was expended to make the lettering exciting. Still, seeing the answers flash on the screen adds a nice element. What's the most exciting moment of any game of *Trivial Pursuit*? When an answer is announced as correct, of course. And having that verification supplied visually is doubly exciting.

Little messages that appear following correct and incorrect answers also add to the fun: "Maybe you'd better study up on that subject," "Gee, you must be smarter than you look," etc. The novelty wears off long before game end, but it's fun all over again with each new group of players.

Not having to put down your soda or quickly down your handful of Doritos to pull and read a card, roll a die, or shuffle a playing piece around a board is an advantage as well. You can stretch out comfortably anywhere within view of the screen, so long as one player sits by the computer and keeps things going. Of course, the inconvenience associated with this convenience should be apparent: most people don't have their computer, disk drive, and monitor set up in their living room. Disconnecting cables and moving hardware is no one's idea of party fun – not to mention putting it all back in order, along with the rest of your house, after your guests leave. You can always play in your computer work room, if you can fit the chairs, and if you don't mind sacrificing comfort. Or you can play *Trivia Fever* without the computer, which it's designed to allow you to do, reading questions out of the thick paperback book packaged with the disk. But then what's the advantage over *Trivial Pursuit*?

Even then, *Trivia Fever* offers a big advantage – in its multiple handicapping facilities. To prevent trivia whizzes from running away with each and every game, you can give them less time to answer their questions (kept track of by the computer; when played manually, by a watch); require them to answer more questions to complete each category; or randomly assign, rather than let them pick, their five of seven categories.

With some imagination, you could adapt these and a dozen other handicapping methods to *Trivial Pursuit* as well. But *Fever,* unlike *Pursuit,* offers the option of Level 1, 2, or 3 questions. Those Level 3's are a very efficient leveler for even the most awesome trivia expert, while almost anyone can at least be competitive at Level 1.

Professional Software, Inc., 51 Fremont Street, Needham, MA 02194 (phone: 617-444-5224).

*–Martin Foster*

***DATABASE MANAGER***
***Mirage Concepts, Inc.***
***Commodore 64***
***Disk; $89.95***

Mirage's *Database Manager (DBM)* gives you a flexible filing system to manipulate just about any type of list or record you want. Store names and addresses, coin or stamp catalogs, book or record collections; or get fancy and automate class records, inventory, or expense accounts. Within the limits of Commodore's disk drive (your complete list of data must be less than about 150,000 characters) and short of needing a true accounting system (with receivables, payables, and general ledger), you can set up almost any type of database.

After reviewing the 174-page manual's excellent tutorial, you can design a custom form on the screen. Although limited to 40 characters in width, it can be about 60 lines long; when you get to the bottom of the screen, the form scrolls up. The process is easy. Position the cursor where you want the label for the first blank to appear, type the label, then type dashes for the length of the blank.

Everything in this program is menu-driven. The top nine screen lines will always tell you where you are and give options for what you can do next. See a mistake? Hit f7 and *DBM* undoes your last step.

Forms are subject to several limits, none of which will restrict you unless you have lots of data. The maximum length of your form (titles and blanks) is 2500 characters. Up to 2000 of those characters can be blanks, or data you enter. Each field, or blank, can be up to 250 characters; and you can have up to 200 fields.

This allows you more freedom than about ¾ of the database programs on the market for the C-64. Needless to say, these limits involve tradeoffs. If you use all 200 fields in your form or record, they cannot each be 250 characters; that's way above the 2000 character limit for single record. Some of your fields can be calculated from the contents of other fields so, for example, *DBM* could automatically enter tax and total cost on an invoice form.

After the form is safely saved on disk, you can enter data. A blank form will appear; just use the function keys to position the



*174 p. manual has excellent tutorial.*
*READER SERVICE NO. 7*

cursor at any blank to enter or to correct data. When you're done, hit a function key to store the data and go to the next record. If you later realize you've made some mistakes, it's easy to recall individual records and correct only those fields which have bad data. *DBM* will accept some incorrect input, such as *$8.9.9,* without warning you.

Next, you'll probably want to sort your records in alphabetical or numerical order based on the contents of a field. You can sort your data only in ascending (A to Z) order, though you can review backward through the file. *DBM* also handles multiple sorts. Say you want your address list in order by zip code, but within each zip code, you want the records in order by last name. Just sort the file twice; first by name, then by zip code. You can do this for as many fields as you want.

*DBM* does not actually rewrite the records in the new sorted order on the disk. It creates a separate file which points to the records in order. The beauty of sorting this way is that you can create separate sort indexes for several fields in the same file. With your address list, you could create a membership list (use the last name sort index) and mailing labels (use the zip code sort index).

When you have all your data entered, finding individual records or groups of records is a snap. Once again, just follow the menus. You can use any arithmetic operators allowed by BASIC. Enter a simple equation (up to 70 characters) like f$(3)="SMITH" AND f$(9)< >"10001" to find all the Smiths who don't live in one zip code in New York City. The f$(3) represents the third field you entered when you built the form. Search equations can actually be quite complex, allowing you to find very specific records. Even the BASIC functions MID$ and VAL are allowed.

The printing capabilities of *DBM* are its weak link. To get more sophisticated features, you will have to buy its companion program—*Advanced Report Generator* for $49.95 (reviewed next issue).

With *DBM*, you have three printing options. You can print each record on a separate form, exactly as it appears on the screen. You can print your data in columns with each record on one row, and headings, like field names, for each column. Or, you can print mailing labels with up to four lines. Unfortunately, the first line of a mailing label can represent only one field from your




*Tell it* **Write Now!**, ***and it will—with instant execution of instructions.***
*READER SERVICE NO. 8*

form, so if you used separate blanks for first and last names, only the last name will fit on the mailing label.

*DBM* allows you to change the format of a form or to add new fields with its Merge Files function. Create a second form, with the new fields included, and *DBM* can automatically move all the data from the old form into the new one. Since both databases will have to exist at one time on the same disk, the maximum size of a file to be reorganized like this is about 75K.

To get around that limit, you could always break your database file into subfiles. Creating subfiles is also a nice way to clean up a database. If you keep class records on your sixth, seventh, and eighth graders, at the end of the school year you can have *DBM* create a subfile of just the sixth and seventh graders—the ones you'll have again next year. There is even a powerful Replace Fields function that lets *DBM* go through the records and automatically advance the grade level of each child who passed.

*DBM* gives you a flexible filing system that is compatible with the form letter functions of most word processors. Its inability to do subtotals limits its possible financial applications, and its limited report formatting will probably force you to buy the companion program. Mirage has created a program that is capable of performing the functions of many of the more-limited programs on the market like recipe filers, automated gradebooks, and mailing list programs.

Mirage Concepts, Inc., 2519 West Shaw, Suite 106, Fresno, CA 93711 (phone: 800-641-1441).

*—Richard Herring*

## *WRITE NOW!*

***Cardco, Inc.***
***Commodore 64***
***Cartridge; $39.95***

Finding a full-featured, inexpensive word processor for the Commodore 64 can be difficult. Finding one that works with tape or disk is harder. Finding one that's fast is even harder. Here it is!

Cardco, Inc. has a habit of seeing a marketing hole and then coming up with the product to fill it. They started out with printer interfaces, then keypads, letter quality printers, and now word and data processing—all aimed at the Commodore 64. With *Write Now!*, they've plugged a hole perfectly.

Because it's on cartridge, the program loads automatically. It takes awhile, but a lot less time than a program on disk or tape. Once it's there, it's quick. Except for the rare occasions of loading or saving text, execution of instructions is instantaneous. Text can be saved to disk or tape. In fact, *Write Now!* is flexible enough even to work with a hard disk system, if one comes along.

It's hard to decide where to start saying nice things about this program. It has all the features of expensive word processors: from page formatting to boilerplate text, reading one file into another, chaining files together, block manipulations, find and replace searches, even conditional paging. With the *Mail Now!* disk program, merge printing can even be done. Headers and footnotes are easy, too.

The only time the program goes to a menu is at print time or to do disk operations. These menus are called by using the Commodore's eight function keys. The default values in the print menu are so well chosen most users won't need to change them.

There are several hidden options in the print menu. One will output the text to the screen as it would appear if printed. This "preview" mode can save a bun-

dle of paper if used properly. Should the user's printer need line feeds in addition to carriage returns, one hidden print option is the auto line feed mode. Another is the ASCII conversion mode. This option converts CBM ASCII to regular ASCII before sending it to the printer. Because these options are only used in special circumstances, they don't appear in the regular print menu.

A separate menu can be called to select tape or disk, the printer device number, and the secondary address.

Disk functions are performed from another menu and do not interfere with the text in memory. This means that if you run out of room on a disk, you can format a new one and use it or delete files until there is enough room on the old disk. It is also possible to rename files from this menu. A file may be saved under more than one name, so the user can create backup files easily. The last filename used is saved as a default, and *Write Now!* automatically scratches the old file before writing the new one, preventing disk errors.

Two of the function keys save and load files. After activating the save function, the program needs to know the file name to store the text in. As mentioned earlier, the last file name entered appears as a default. This is changed simply by typing in a new file name. The load command works similarly.

The documentation is excellent. The table of contents is very specific. The index is well done. And the positive, encouraging tone is marvelous. The authors don't expect the new user to be totally fluent in word processing terms. The text of the manual is informative without being condescending. The keyboard overlays provided make it easy to figure out which keys to press to accomplish any goal. Should the user forget a command that isn't on the keyboard, pressing the function seven key calls up a group of Help! menus.

Like many of the top rated programs, *Write Now!* uses dot commands to perform many tasks. These would be the primary reasons a user would need the Help! menus. All the commands are listed alphabetically in the index. These listings contain a brief (two or three words) description of what the dot command does. For instance, .SP is the command to send a special code to the printer. The index listing reads

**.SP (SEND PRINTER CODE). . . . . 55**

Page 55 contains a one-paragraph explanation of the command and how it works. All of the dot commands are explained this way. One trip through the manual should get the average user familiar with the layout. With the depth of the indexing, it shouldn't be hard to locate any particular command within a few minutes.

To really help the novice word processor user, Cardco has even included a listing of the special codes needed by the most commonly used printers. These special codes allow the user to turn underlining, boldface, emphasis, expanded print, italics, subscript and superscript, and/or correspondence mode on and off. Not all printers can perform all these functions, but the *Write Now!* manual includes tables with the codes for the most popular printers currently in use. One table has Epson, Star, Okidata, Prowriter, C-Itoh, NEC 8023, Axiom, and Commodore dot matrix printers. Another has the Cardco, Brother, Smith Corona, Silver Reed, and C-Itoh typewriter quality printers.

One especially nice dot command is .BB (building block). This command lets the user string together a set of files in such a way as to create a new document. A letterhead and signature block need only take up a small amount of disk space. Instead of having these items take up space in every letter written, they can be given their own file names and stored on the disk. By using .BB and the filename, the letterhead or signature block can be inserted wherever needed using only a few characters of disk or tape space.

Although they can be assigned in the print menu, some parameters are also dot assignable. Line spacing, left margin, text width, top and bottom margins, page length, and page numbering can all be set using dot commands. Adjusting them this way can be easier than changing the print menu.

My own experiences with using the program have been positive. It has done everything it should. The printouts look good. For $39.95, the user definitely gets his money's worth. Recommending this program comes easy to me. A serious word processor, indeed.

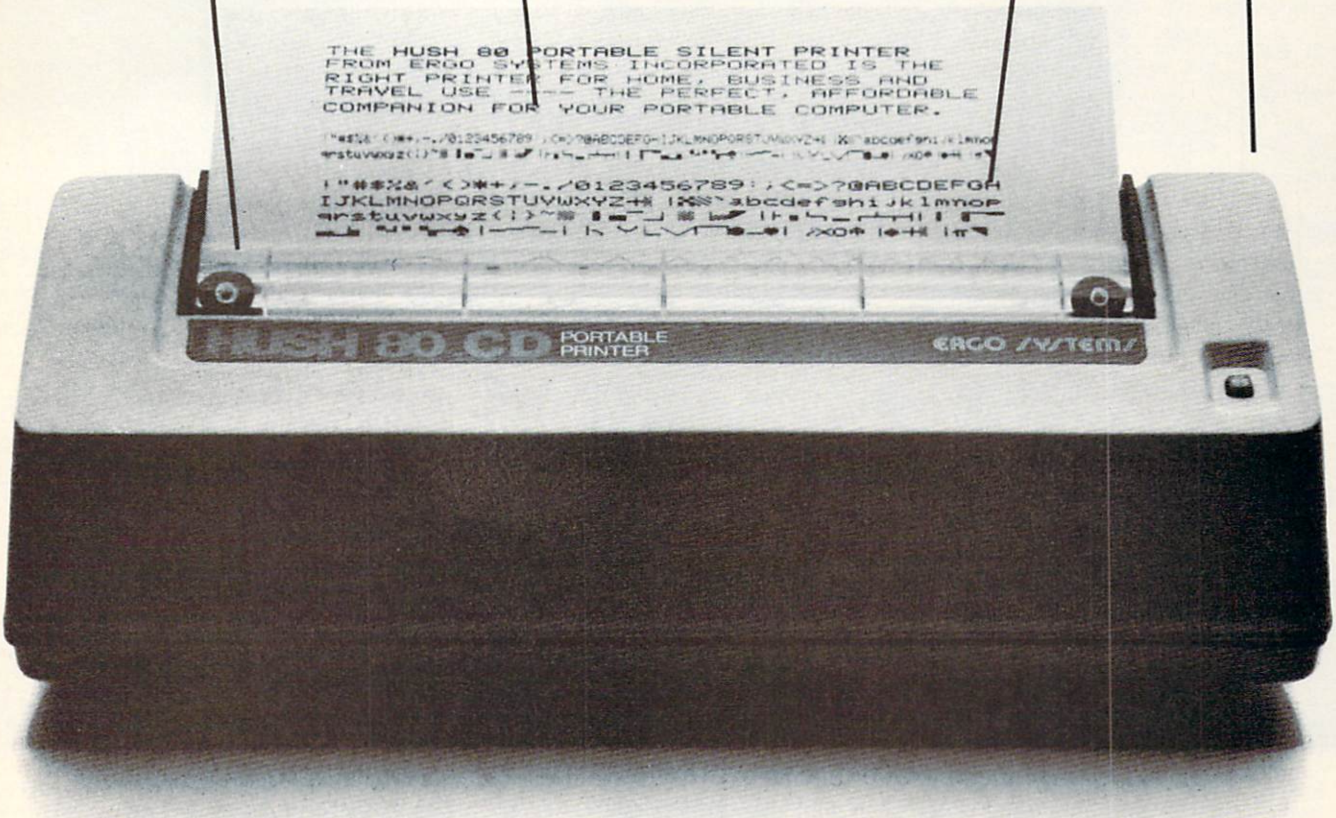
Cardco, Inc., 300 S. Topeka, Wichita, KS 67202 (phone: 316-267-3807). *—Cheryl Peterson*

## *GRIDTRAP*
***Livewire Software***
***Commodore 64***
***Disk, cassette; $24.95***

The last British invasion on American culture was led by the Beatles, 20 years ago. It drastically changed the music American teens listened to. The present British invasion, of game software, has shown no potential for such standard-shifting: just good old games in the good old American mode.

*Continued on page 46*

# BIG ON FEATURES.




**$139.99**

# SMALL ON PRICE.

You read that right. A full 80 column, 80 character-per-second, Commodore-compatible dot matrix thermal printer, complete (with a 90 ft. paper roll, interface cable and power pack), ready-to-run— at 800 words per minute — for only $139.99. But your wallet isn't the only thing the HUSH 80 CD is compatible with. It's compatible with all Commodore models; including the Commodore 64, the VIC 20, and the new Commodore 264.

And you'll find the HUSH 80 CD printer is big on lots of small things too. It weighs about one-half the amount of it's nearest competitor (it's compact size fits conveniently into a briefcase). And it's extremely quiet, with fewer working parts, which means less noise and less to go wrong.

Ask your local computer store or dealer for the HUSH 80 CD today. And if he doesn't carry it, call us at 415/322-ERGO. And find out where to get the little printer that does the big jobs.

**HUSH 80 CD™ PRINTER**

**LESS PRICE, LESS NOISE, LESS SIZE.**

ERGO SYSTEMS, INC. 26254 Eden Landing Road, Hayward, CA 94545

# Making music was never this much fun.™




Imagine yourself in a professional recording studio, manipulating richly musical sounds from any of sixteen different instruments. Imagine creating your own musical compositions, just like a recording superstar. Imagine starting as a novice and becoming a musical master. Now the future of musical entertainment is as close as your home computer... Introducing MELODIAN™ **THE COMMODORE-64™ COMPATIBLE MELODIAN KEYBOARD** plugs right into







your home computer...and then the fun begins. The menu driven display screen lets you experiment with infinite varieties of instrumentation and orchestration all with only four simple command keys. And the colorful user-friendly graphics invite you to learn the basics of music at your own pace.



**THE MELODIAN CONCERTMASTER™** software package allows you to record and mix *multi-track* music just like a professional recording engineer. Put down the bass track and get it just right. Then add guitar, brass or woodwinds. Experiment with MELODIAN'S unique ADSR synthesizer window to custom design your own new instrument sounds...and before you know it, you have a completely new musical arrangement. Or you can study the intricacies of prerecorded musical pieces, note for note ...and then rearrange them to your heart's content.

**THE MELODIAN RHYTHMMASTER™** software package starts with your own natural sense of rhythm...the building block of music...and takes it to perfection. Meet the challenge of RhythmMaster as you learn the basics of rhythm and pitch. Track your progress until you make it all the way to virtuoso.

**AFFORDABLE BREAKTHROUGH TECHNOLOGY...ONLY $199.90.** The MELODIAN system is the result of breakthrough computer design innovations, invented by the same team who developed high-tech music systems for recording superstars. That's why for less than $200, you can start with the Melodian Keyboard and ConcertMaster and start making music right away. RhythmMaster can be added for only $39.95. And other exciting prerecorded programs will be available soon. MELODIAN products are available at computer and music stores. Or you can call Melodian directly at 1-800-MELODIA and charge it on all major credit cards. All Melodian products have a 15 day money-back guarantee.

MAJOR CREDIT CARDS ACCEPTED

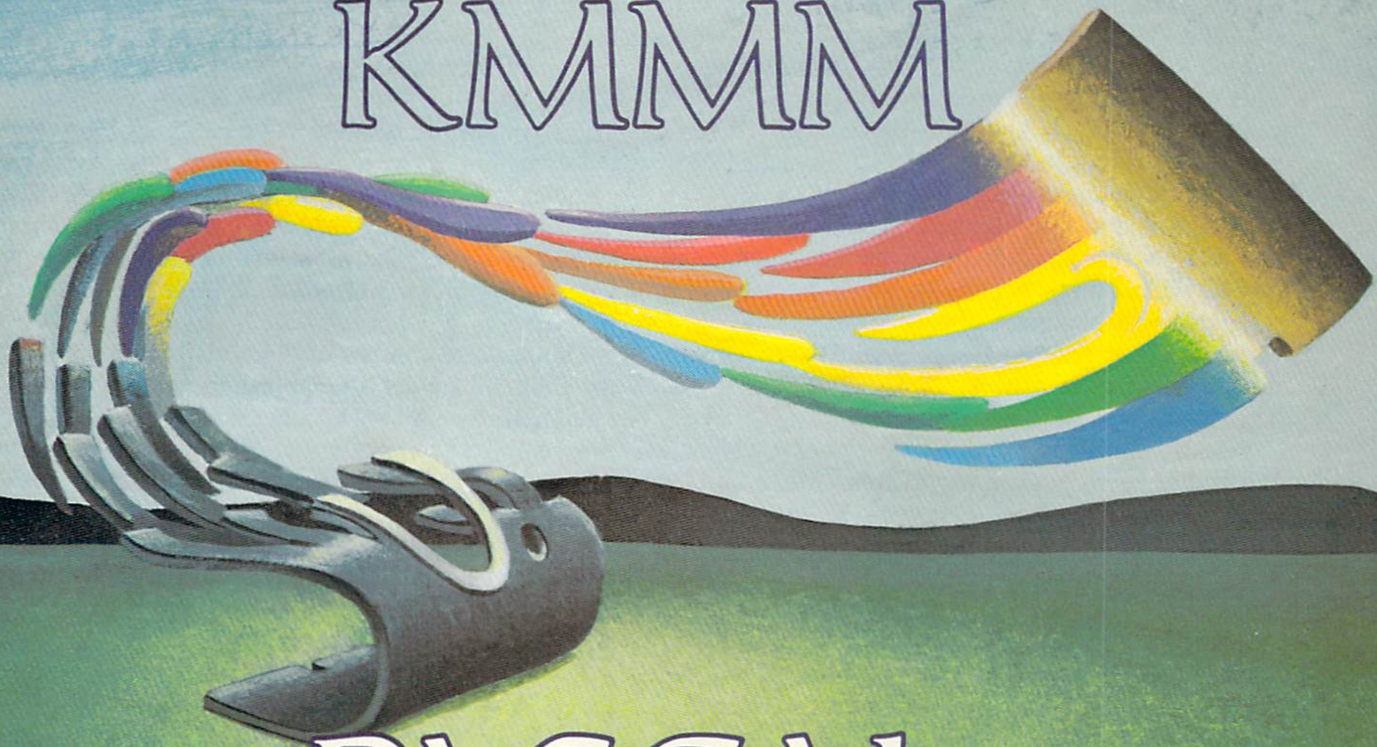

MELODIAN™

The new, fun way to learn, play and compose music.

CALL 1-800-MELODIA
IN NEW YORK STATE CALL
COLLECT 1-212-513-7330

Reader Service No. 50

# A USER'S GUIDE TO KMMM PASCAL



When the Commodore 64 is turned on, the programming language available to the user is BASIC. While most users will be content with the status quo, for many the question of "what else?" will arise. This event will come to pass for even the most mildly dedicated programmers for several possible reasons. It may simply be curiosity about some of those other high level programming languages. Or perhaps, it is a mastery of BASIC that initiates the urge for something "better" or "different".

In the case of Pascal the motivation to learn a different language may even be external. In many schools throughout the country, Pascal is the language of choice for educators. The ability to practice on your own machine, whenever you like, is a powerful motivator.

*KMMM Pascal* by Wilserv Industries has been available for the PET computers, and more recently for the Commodore 64. It is a fine product, well worth its $99 retail price. *KMMM Pascal* is well known within the PET community. The general lack of promotional activity, and some modesty on the part of the developer, have kept *KMMM Pascal* in the background for recent Commodore 64 devotees.

By Morton Kevelson

## *ABOUT PASCAL*

Though we do not have the space to provide a tutorial on Pascal, we will mention some of the language's salient features. It is a structured language. This means that Pascal programs are written in logically contained units known as procedures. These procedures act as building blocks, being called by the main program as needed. Although the GOTO command is available, its use is somewhat restricted and actively discouraged. In comparison, GOTOs in BASIC will at times hopelessly tangle the logic of a program. Variables must also be systematically defined, in comparison to BASIC's willy nilly assignments. These rules may seem restrictive at first. Their usefulness becomes apparent with experience.

Pascal, which was originally developed by Niklaus Wirth, has been standardized to some extent by *The Pascal User Manual and Report* by K. Jensen and N. Wirth (Springer-Verlag). It has achieved favor among educators in that it promotes good programming habits—although, some users point out, its structured nature makes it easy to grade. The end result is the same.

## *FEATURES OF KMMM PASCAL*

*KMMM Pascal* is a subset of Jensen and Wirth Pascal with extensions. It is not our intent to report on all these details at this time. For those readers who are familiar with or have worked with Pascal, we have provided a complete table of all of *KMMM Pascal's* reserved words and pre-defined identifiers. Take careful note of the footnotes in the table. These indicate the features of "Standard Pascal" not presently supported by *KMMM Pascal.* We have also marked those features of *KMMM Pascal* which are extensions to the standard.

We must also emphasize that *KMMM Pascal* is a dynamic package, fully supported by its developer. This means that upgrades are regularly issued. We first became acquainted with Version IV.1 of *KMMM Pascal* just about a year ago. This report is based on version IV.6B. In the course of these revisions, several of the missing standard features were implemented. A number of extensions were added, several old "bugs" were eliminated, and some new ones were discovered and subsequently swatted. It is likely, given the normal publication delays, that this version of *KMMM Pascal* will be obsolete by the time you read this. Some of the shortcomings in the table will most likely be eliminated. In all probability, a number of extensions will have been added.

Upgrades are offered by mail to all registered users. Although the upgrade fee may be as high as $7.50, all the upgrades we have received have been for a very nominal $1.00 if we supplied our own disk, $3.00 without our own disk. Be sure to return your registration certificate. This is one package where registration really means something. Do take advantage of the $2.00 incentive to register your package. For this nominal fee, you will receive a disk full of *KMMM Pascal* demonstration programs submitted by other users.

The output of *KMMM Pascal* is fully compiled. The resulting file is 100% 6502 machine code, with the exception of the initiating line number and SYS command. The programs produced by *KMMM Pascal* can be loaded and saved the same as any other BASIC program. The programs will run completely independent of the rest of the *KMMM Pascal* package.

The package consists of three major components, the Editor/Compiler, the Compiler, and the Translator. Their use may seem a bit cumbersome at first, especially when compared to the operation of BASIC. This is somewhat deceptive, as a little familiarity with the system will overcome most inhibitions. We will now briefly describe the operation of each component and their interactions.

## *THE EDITOR/COMPILER*

Unlike BASIC, there are no line numbers in a Pascal program. Instead, a Pascal program is written very much like any other text. Of course there are rules for proper syntax which must be followed. The Editor/Compiler is nothing more than a custom word processor for the creation of *KMMM Pascal* source files. It has two different modes of operation.

The command mode is primarily intended for the maintenance and creation of *KMMM Pascal* disk and tape files. The operation seems a little strange at first. However, with familiarity, the command sequences become very logical and convenient. The first operation is to define a control character. The British Pound symbol (£) is recommended. The control character is used to separate commands with two consecutive characters designating the end of a command sequence. For example, the sequence:

```
GR0:PLABEL£Y£GC££
```

would open a sequential file named "PLABEL" for reading, "Y"ank the file from the disk into the text buffer and close the file. The command sequence would not be acted on until the second consecutive £ symbol was typed.

In a similar fashion, there are commands which allow the creation of new text files as well as editing of existing text files. The latter are rather cumbersome. For this reason, the second Editor mode is provided.

In the window, all the facilities of the Commodore full screen editor are available. Simply position the cursor anywhere on the screen using the cursor control keys and bang away. The INST/DEL key functions as expected. In addition, the function keys are enabled to perform insertion and deletion of lines as

well as activating an insert mode.

*KMMM Pascal* is a compiled language. With one exception, there is no way to write some code and immediately test it as with BASIC. When the Compiler portion of the Editor/Compiler is executed, the Pascal source file in the buffer is checked for syntax errors. Corrections can be made immediately afterwards. Operational testing of the program still requires full compilation.

An abbreviated version of the Editor without the Compiler is provided. This version lacks the syntax checking feature but takes up less space in the computer's memory. This allows longer source files to be created.

## THE COMPILER

Once a *KMMM Pascal* source file is created and saved to disk, the Compiler is invoked. The compilation process reads the selected source file, checks for syntax errors, and generates an intermediate p-code which is left in the computer's memory. Version IV.6B has the option to save the p-code to disk, but the package has no way to work with the resultant file. Future updates can be expected to include this feature.

## THE TRANSLATOR

The final step in the process involves the Translator. This can be loaded and executed by the Compiler when its job is done. The Translator converts the p-code stored in RAM into object code, or 6502 machine code, for direct execution. Upon completion, the user is left with a fully compiled machine language program in RAM. This can be SAVEd, LOADed, and RUN just like any other program. An eight-kilobyte run time package, which is appended to the resultant code, becomes a part of the final program. Thus fully compiled *KMMM Pascal* programs can be created and distributed within the package's copyright agreement. When a compiled program is RUN, a message briefly appears indicating the Wilserv copyright to the run time package.

## THE USER MANUAL

This is the weakest link in the entire package. The manual is supplied in two main parts: the Compiler/Translator Reference Manual and the Editor Reference Manual. The basic manual actually applies to version III.9 of *KMMM Pascal.* A Version IV.1 Upgrade manual is appended to the package. To further add to the confusion is a rather lengthy *Errata* file on the disk. This documents all changes and modifications since the release of version IV.1. An *Addendum* file on the disk lists all extensions added to *KMMM Pascal* since the Upgrade Manual. Both of these files have the effect of modifying both the Reference Manuals and the Upgrade Manual.

Fortunately, the manual is in the process of being revised. An integrated version should be available by the time this report is published.

The present manual is not for beginners. Even though a comprehensive tutorial is given, the novice will have to spend some time carefully trying out the procedures. You will not be able to learn Pascal from the manual. All Pascal discussions deal with the unique features of *KMMM Pascal* as compared to the standard Pascal. (Several Pascal reference books are listed in the manual, as well as at the end of this report.)

## THE USER LIBRARY

Several *KMMM Pascal* demonstration programs are provided with the distribution diskette. Additional programs are on the User Library, which you will receive when the registration form is returned. These files have to be compiled and translated before they can be RUN. By all means do so. These programs include a number of useful utilities as well as demonstrating the implementation and extensions unique to *KMMM Pascal.* The programs actually form part of the documentation supplied with the package.

## ADDITIONAL FEATURES

Several utilities are provided on the distribution disk. Among these is a custom version of the DOS wedge. Support is also provided for the MSD CIE

IEEE-488 interface. Using this interface with an IEEE-488 disk drive, such as the CBM 4040 or the MSD SD-2, can more than double the speed of disk operations.

The package includes a "security key" to prevent unauthorized distribution of the program. This limits the use of the package to one computer at a time. The security key, which plugs into the cassette port, must be present for operation of the Compiler. All other parts of the package, including the run time library, will function without the security key. The cassette port connector is extended by the security key to allow simultaneous operation of a cassette recorder.

## CONCLUSIONS

*KMMM Pascal* is a comprehensive implementation of "Standard Pascal" with extensions, for the Commodore 64 and PET computers. User support with regular upgrades is excellent. Students of Pascal should find *KMMM Pascal* quite sufficient for their needs.

The fully compiled output of the package will run several times faster than equivalent programs written in BASIC. The inclusion of the stand-alone run time library will be of interest to software developers.

KMMM Pascal
Wilserv Industries
P.O. Box 456
Bellmawr, New Jersey 08031
(609) 227-8696

## REFERENCES


1. *Pascal For BASIC Programmers,* Charles Seiter/ Robert Weiss. Addison Wesley, $10.95.
2. *Introduction to Pascal Including UCSD Pascal,* Rodney Zaks. Sybex, $16.95.


### KMM PASCAL RESERVED WORDS

| | | | |
|---|---|---|---|
| AND | END | NIL | SHL+ |
| ARRAY | FILE | NOT | SHR+ |
| BEGIN | FOR | OF | THEN |
| CALL+ | FUNCTION | OR | TO |
| CASE | GOTO* | PACKED | TYPE |
| CONST | IF | PROCEDURE | UNTIL |
| DIV | IN | PROGRAM | VAR |
| DO | LABEL* | RECORD | WHILE |
| DOWNTO | MEM+ | REPEAT | WITH* |
| ELSE | MOD | SET (1) | |

### PREDEFINED IDENTIFIERS

| | | | |
|---|---|---|---|
| **CONSTANTS:** | | | |
| FALSE | TRUE | MAXINT | |
| **TYPES:** | | | |
| BOOLEAN | INTEGER | STRING+ | TEXT |
| CHAR | REAL | | |
| **FUNCTIONS:** | | | |
| ABS | EOF | MIDSTR+ | RND+ |
| ANDB+ | EOLN | NOTB+ | ROUND |
| ARCTAN | EXP | ODD | SIN |
| CHR | GETKEY+ | ORB+ | SQR |
| CONCAT+ | INKEY+ | ORD | SQRT |
| COPY+ | LEFTSTR+ | POS+ | SUCC |
| COS | LENGTH+ | PRED | TRUNC |
| DELETE+ | LN | RIGHTSTR+ | |
| **PROCEDURES:** | | | |
| CLOSE+ | PACK* | READLN | UNPACK* |
| EXIT+ | PAGE* | RENAME* | WRITE |
| ERASE+ | PUT | RESET | WRITELN |
| NEW | READ | REWRITE | |
| **FILES:** | | | |
| INUPT | OUTPUT | | |

* Not implemented in version IV.6B
\+ ***KMMM Pascal*** extension to standard Pascal
1 Recognized but not implemented

# REVIEWS

*Continued from page 40*

*Gridtrap* is the best of three Livewire releases made available in this country by 3R. It offers no substantial twist on the maze game theme. But if maze games happen to be your cup of high tea, *Gridtrap* can hold its own with most.

The graphics are simple: a 150-box (10 X 15) grid which Mr. Livewire travels across, erasing boxes as he goes. The goal is to defuse time bombs that appear onscreen with a 30-second fuse. As soon as Mr. Livewire defuses one (by stepping on it—let's see the bomb squad try that) another appears, up to five. If the countdown reaches 15, a second bomb appears onscreen simultaneously with the first.

You'll have plenty of time to reach the bombs if you utilize your ability to scroll Mr. Livewire off the left of the grid and back on the right, or off the top and on the bottom, and vice versa, vice versa. Boots appear onscreen which can stomp Mr. Livewire, but they don't chase him very actively at the lower levels. Skulls, deadly to the touch, occupy some squares, but they stay put and can easily be avoided. Your main enemy is yourself, your main danger lying in erasing so many boxes in your travels that you find yourself unable to reach a ticking bomb. Even then, you can scroll rows of boxes toward you, but occasionally you will reach an impasse.

The game includes such touches as a pause feature and the ability to define your own keys for keyboard play. If you want to skip entering your initials and seeing your place in the Hall of Fame after each round, tough.

In its basic gameplay, *Gridtrap* is like many other maze games. Unlike many other maze games, it has wit, requires strategy, and reflects some care on the part of the designer.

Livewire Software, c/o 3R Import and Export Corporation, 731 James St., Suite 405, Syracuse, NY 13203 (phone: 315-475-2224). *—Greg Fried*

# One Letter is a Thousand Pictures

## FAST GRAPHICS WITH CUSTOM CHARACTER SETS

When you're creating a game that moves objects around on the screen, it's important to make those objects look like something. If your game is to be memorable and fun to play, it must give the player an experience that can't be forgotten—and the single most powerful tool you have in creating such an experience is the power of the computer to make pictures on the screen.

But making pictures isn't as easy, unfortunately, as xeroxing something. You have to construct your visual images out of little dots of color on a television screen. Think of it—you have to account for every single one of the dots on the TV screen. That's a lot of dots. That's why graphics programming is usually regarded as advanced programming.

One system for creating pictures is to use bit-mapped graphics—but on the VIC and 64, that means drawing all your pictures with machine language—or with endlessly slow PEEKs and POKEs.

You can also create pictures with sprites (on the 64)—and we will, a few months from now—but again, sprite movement requires either machine language or PEEKs and POKEs, so that any kind of speed is difficult. Wouldn't it be nice if your computer contained a set of hundreds of different building blocks that you could put together quickly on the screen to make many different pictures?

### BUILDING BLOCKS YOU DESIGN YOURSELF

Good news. Your computer has exactly the set of building blocks you need. You can put the blocks together to make pictures at machine language speed using the string functions and PRINT. When you power up your computer, all those blocks look like letters and numbers and symbols and graphics characters—but unlike the building blocks you played with as a kid, these blocks don't have to stay the same shape. You can mold them to any shape you like.

Both the Commodore 64 and the VIC allow you to create your own building blocks by redesigning the patterns of the character set. You can use this feature simply to make letters and numbers that have a different look from the built-in character set—that's what many word processors for the 64 do.

However, there's no law that says the characters you create have to look like standard characters at all. Your computer doesn't care if an A looks like an A. To the computer, the letter A is just ASCII code 65, or screen code 1. No matter what shape you give to character 65, when the computer meets a command like PRINT "A", it will put character 65 on the screen by storing screen code 1 in screen memory. If character 65 happens to look like a letter A, then that's good for the human being trying to read it; but it can look like a radish for all the computer cares.

That means character 65 can be part of a castle wall, or part of a boat, or an entire spaceship, or an animal, or a little human figure. And when you use animation techniques to move characters around the screen, the player of your game will see the picture, not the letter A.

If you've been following this series from the be-

By Orson Scott Card

ginning, you'll remember that in an early article we already used custom characters. We have moved boats back and forth across the screen in a harbor, and even though the computer thought it was PRINTing strings like "ABC" and "DEF", what the player saw was ocean liners and little launches, lighthouses and harbor buoys. However, I offered no explanation at the time—because custom characters require an entire article to explain. This is the article.

At the end of this article, after all the explanations are over, there is a program that loads a group of custom characters into memory and loads them into strings, so you can PRINT them on the screen anywhere you want. The different objects won't all make sense together, but it will give you a chance to see how custom characters can be used to create figures and background displays. And you can replace any of my characters with characters you design yourself, using the techniques in this article.

### *THE SHAPE OF A CHARACTER*

If you look closely at your television or monitor, you can see that the letters are composed of little dots. To save you from developing eyestrain, I won't ask you to count them.

Each character on the screen is composed of 64 dots on an 8 by 8 matrix. That means that there are eight rows composed of eight dots each. The dots are slightly higher than they are wide, so that instead of forming a square, the character matrix is a tall rectangle.

Each dot in the character can display one of two possible colors—the background color and the foreground color. The background color is the same for the whole screen, and consists of the color stored at location 53281. The foreground color can be selected individually for each character, and the color data is stored in color memory.

When your computer powers up, the background color is dark blue and the foreground color is light blue. So the character matrix for, let's say, the letter I might look like this, with X's representing dots of foreground color and hyphens representing dots of background color:

```
- - - - - - - -
- - X X X X - -
- - - X X - - -
- - - X X - - -
- - - X X - - -
- - X X X X - -
- - - - - - - -
- - - - - - - -
```

Notice that this character shape leaves blank rows at the top and bottom, and blank columns to the left and right. This is because each character needs to be clearly distinct from the character next to it. The computer doesn't leave spaces between character shapes—the characters run right into each other. If the character shape doesn't provide for blank space between letters, there won't be any, and goodbye legibility!

However, when you're designing your own character shapes, you may actually want the shapes to run together. For instance, you may want to design a ship that is wider than a narrow character matrix will allow. So you can design the shape in two characters, and then PRINT them side by side on the screen:

```
- - - - - - - -   - - - - - - - -
- - - - - - X X   - X X - - - - -
- - - - - - X X   - X X - - - - -
- - - - X X X X   X X X X X - - -
X X X X X - X -   X - X - X X X -
- X X X X X X X   X X X X X X X -
- - X X X X X X   X X X X X X X -
- - - - - - - -   - - - - - - - -
```

Individually, these shapes would mean nothing. Together, though, they make a picture. And because there are no blank lines between the two halves of the shape, they combine on the screen into a single continuous image.

Together, then, these two characters combine to make a single complex character. Within a program, you would link these characters in a single string, to be PRINTed together on the screen. If the left side of the ship had taken the place of "A" and the right side of the ship had taken the place of "B", then you would create a string variable like this:

```
A$="AB"
```

Once this command was executed, whenever you issued the command PRINT A$, the string "AB" would be placed on the screen—only instead of seeing AB, you'd see the ship.

You might notice that in designing this character I left a blank row at the top and bottom of the characters, and a blank column on the rightmost edge of the righthand character. This means that if two of these complex characters were PRINTed next to each other or one above the other, there would be a space between the ships.

So it's as important to plan how the character

shapes will combine with each other on the screen as to plan the shape within the character matrix.

## *FROM SHAPES TO NUMBERS*

How do we get our shape designs into the computer? After all, computers only work with numbers, not pictures.

It is not just a coincidence that each row in the character matrix has eight dots, and each byte in memory has eight bits. The character set is stored in memory in the same order as the screen character codes (not the ASCII codes). That is, the first character is @, which has a screen code of 0; the second character is A, with a screen code of 1; then B (2), C (3), and so on up to screen code 255, which is the reverse-video pattern of screen code 127.

Each character matrix consists of eight rows of eight dots, so to contain the pattern in memory requires eight bytes of eight bits each. Therefore the character set takes eight bytes for each of 256 characters (0-255), for a total of 2048 bytes.

You can always find the pattern for any character if you know the character's screen code and the starting address of character memory. Let's say that character memory starts at location 14336. We want to find the pattern for the letter M. The screen code of M is 13. So the pattern for M begins at 8*13 bytes (104 bytes) from the start of character memory.

To find the pattern—*if* character memory begins at 14336—we could use this program line:

```
PT=14336+8*13
```

The variable PT now holds the address of the first byte of the character pattern for the character "M". That byte will hold the pattern for the top row of the character. PT+1 holds the pattern of dots for the second row of the character, and PT+2 holds the pattern for the third row. The last, or eighth row, is contained in byte PT+7. The next character pattern, for the character "N", begins at PT+8.

How do we convert dots into numbers that can be POKEd into character memory?

Let's start with the left side of the ship pattern we made.

```
- - - - - - - -
- - - - - - X X
- - - - - - X X
- - - - X X X X
X X X X X - X -
- X X X X X X X
- - X X X X X X
- - - - - - - -
```

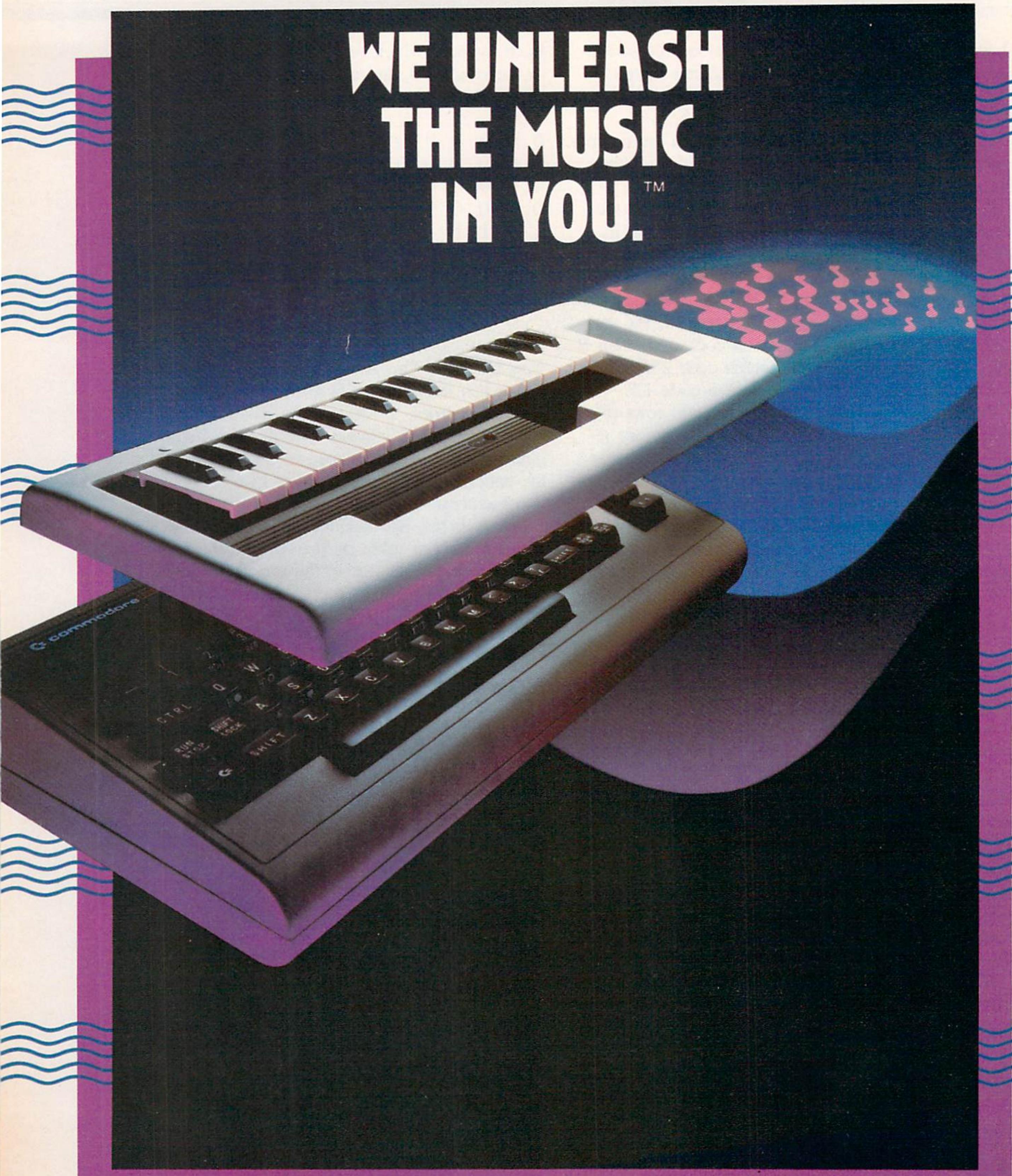
Each hyphen, which represents the background color, becomes a 0, and each dot of foreground color (X) becomes a 1:

```
0 0 0 0 0 0 0 0
0 0 0 0 0 0 1 1
0 0 0 0 0 0 1 1
0 0 0 0 1 1 1 1
1 1 1 1 1 0 1 0
0 1 1 1 1 1 1 1
0 0 1 1 1 1 1 1
0 0 0 0 0 0 0 0
```

These are now eight bytes, or eight numbers in binary notation. This is what the computer actually sees when it PEEKs at bytes in memories—rows of ones and zeroes.

However, BASIC works with decimal numbers, not binary numbers, so we need to convert each byte in the pattern to a decimal number in order to POKE it into memory.

Binary bytes, like decimal numbers, are set up with the highest value on the left and the lowest value on the right. Just as in the number 321 the 3 represents hundreds, the 2 represents tens, and the 1


WE UNLEASH
THE MUSIC
IN YOU.™

# The INCREDIBLE MUSICAL KEYBOARD™

## transforms your computer into an exciting musical instrument.



### The Incredible Musical Keyboard can do more than turn your Commodore into a music synthesizer.

Deep within every Commodore 64 lurks the unexpected. A versatile music synthesizer. A music video machine. A three track recorder. And a way to create music that's so easy you'll be playing songs within minutes.

Just place the Incredible Musical Keyboard over the computer keyboard, load the included music software diskette, and you have all you need to make piano, guitar, synthesizer and electronic sounds.

### Everything you need to create music is included . . . whether you're a beginner or an accomplished musician.

If you can press a few keys, you have all the skills it takes to create and play music with the Incredible Musical Keyboard.

Just beginning? Attach the included ABC key stickers to the keys and match the letter in the note with the letter on the key and . . . PRESTO! . . . you're playing music instantly. It's easy. It's fun. It unleashes the music in you . . . even if you don't read music.

You already read music? Great! We've included a Melody Chord Songbook featuring hits from today's top recording artists. Or play songs from your own sheet music.

### The Incredible Musical Keyboard works with other exciting Sight & Sound Music Software programs, too!

Get the **KAWASAKI SYNTHESIZER** and transform your Commodore 64 into a sophisticated music synthesizer and sound processor. With one keystroke, you can control waveforms, vibrato, sweep and other special effects. Let your imagination soar with the **KAWASAKI RHYTHM ROCKER** creating and recording your own songs while you control eye-popping color graphics that explode across the screen. Create your own music with the **MUSIC PROCESSOR**, a word processor for music. Then record it, edit it and play it back. Use the 99 preset instrument and special effects sounds for endless new creations. Or travel through a complete tutorial on music synthesis with the newest Sight & Sound music program, **3001 SOUND ODYSSEY**. After you've learned the basics of music synthesis, you can use the complex music synthesizer included in the program to compose your own music, create your own sound effects or simply experiment with the hundreds of pre-programmed sounds.

### From the Incredible Musical keyboard to software that plays your favorite hits, we've got them all.

Look for the Incredible Musical Keyboard and the complete line of Sight & Sound music software at your local computer software outlets and music stores. And unleash the music in you.




For more information and a free catalog, phone 1-414-784-5850 or write P.O. Box 27, New Berlin, WI 53151.

SIGHT & SOUND
Music Software, Inc.

**WE UNLEASH THE MUSIC IN YOU.™**

**Reader Service No. 17**

represents ones, so in binary notation the leftmost bit is the highest number and the rightmost bit is the lowest number.

In binary notation, the rightmost or lowest bit represents 1. This means that if the bit is *on,* or contains a 1, then the bit has a value of 1. If the bit is *off,* then it has a value of 0. Each bit to the left is double the value of the bit to the right of it, like this:

```
128 64   32   16    8    4    2    1
```

To calculate a decimal value, add the values of the bits that are *on* (1) and simply ignore the bits that are *off* (0). Here is the eight-byte pattern for the left side of the ship. Below each digit of the binary number is the decimal value of that digit. Then, to the right, you'll find the sum of the values of the digits—which is the decimal value of that byte. If you POKE that decimal number into memory, the computer will find that binary row of zeroes and ones:

| binary numbers | | | | | | | | decimal value |
|---|---|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| no bits on | | | | | | | | 0 |
| 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | |
| | | | | | | 2 | +1 | 3 |
| 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | |
| | | | | | | 2 | +1 | 3 |
| 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | |
| | | | | 8 | +4 | +2 | +1 | 15 |
| 1 | 1 | 1 | 1 | 1 | 0 | 1 | 0 | |
| 128+64 | | +32 | +16 | +8 | | +2 | | 250 |
| 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | |
| | 64 | +32 | +16 | +8 | +4 | +2 | +1 | 127 |
| 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | |
| | | 32 | +16 | +8 | +4 | +2 | +1 | 63 |
| 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | |
| no bits on | | | | | | | | 0 |

These decimal numbers—0, 3, 3, 15, 250, 127, 63, 0—are the pattern for the left side of the ship. If we are putting this pattern in character memory in place of the character "M", we could use this short program:

```
10 CP=14336+13*8
20 FOR I=CP TO CP+7:READ A:POKE I
,A:NEXT
30 DATA 0,3,3,15,250,127,63,0
```

As you will see in the example program on page 75, you can set up as many characters as you want—up to 256 of them—and POKE them into character memory. The only rule is that each character's pattern starts at *charactermemory* + 8 * *screencode,* and consists of eight bytes.

You might also want to use a formula to determine the screen codes. This program does exactly what the above program did, except that you don't have to figure out the screen code where you want to POKE the pattern. Now you decide which character you want the new pattern to replace. Then you begin the DATA statement with the character. The program automatically calculates the screencode:

```
10 CM=14:CS=CM*1024
20 READ C$:C=ASC(C$):IF (C>63 AND
 C<96) OR (C>159 AND C<192) THEN
C=C-64:GOTO 40
30 IF C>95 AND C<128 THEN C=C-32:
GOTO 40
40 CP=CS+8*C
50 FOR I=CP TO CP+7:READ A:POKE I
,A:NEXT
60 DATA M,0,3,3,15,250,127,63,0
```

Because the DATA statement in line 60 begins with M, the program will calculate the screen code of M (in lines 20 through 30) and set CP to the starting address of the pattern for that character. You could replace the M with any other *non*-inverse character, and it would then put the pattern in the place of the new character. (This formula doesn't handle inverse characters.)

One thing to be careful about: If you POKE a new character pattern in place of the space character " ", or CHR$(32), the whole screen will automatically be filled with that new character pattern. This is because a blank screen is filled with screen code 32, the blank character, which normally consists entirely of zeroes. If you change it so that there are any ones in the binary pattern, those *on* bits will show up as foreground color dots, and instead of a blank screen, you'll get a very busy screen indeed.

## *LOCATING CHARACTER MEMORY*

Even though we have designed some characters, translated them to binary and then decimal values, and POKEd them into memory, we *still* haven't set up a new character set.

That's because we have to tell the Commodore 64 operating system where to find the new character memory. (The two computers handle video memory

completely differently; we'll start with 64 memory handling.)

For video display, the Commodore 64 divides RAM into four sections, or blocks of 16K. The VIC II chip, which controls video in the 64, can only read one of these blocks at a time. So all the information for video needs to be located in the same 16K block of memory.

When your Commodore 64 powers up, the VIC II assumes that the video block is the first 16K of memory, from address 0 to address 16383. Screen memory is at 1024, which is within this block. Character memory, however, is in the character ROM chip, which is at 53248. This is outside the video block, but that's OK–the Commodore 64's internal hardware fools the VIC II into thinking that the ROM chip at 53248 is really at 4098–48K lower in memory. You don't have to worry about that–whenever the Commodore 64 is taking its character set from character ROM, the operating system does all the housekeeping.

However, *you* can't fool the operating system that way. You have to locate your custom character set within the video block. You definitely don't want to put it at 4098, however, because then you'd have no room at all for your BASIC program. That's because the BASIC program *also* begins in that first 16K block of memory. So to save space for the BASIC program, you need to locate character memory as high as possible in the 16K video block.

Character memory must begin on a 2K boundary. This is because a whole character set uses 2K. (The 64's character ROM uses 4K, but that's because the built-in ROM contains two entire character sets–you flip back and forth between them when you press SHIFT-COMMODORE.)

So your character set has to be located at 0*1024, 2*1024, 4*1024, 6*1024, 8*1024, 10*1024, 12*1024, or 14*1024. (Remember that 1024 equals 1K.) You can't put character memory at 0K or 2K because those address ranges are used by the operating system and your own BASIC program. In fact, to give yourself as much room as possible, you'll usually put it at 14K, or 14*1024.

In the above examples, that's where we did put character memory.

Now that we've decided where to put our custom character set, we still have to tell the Commodore 64 where to find character memory. We don't have to tell the operating system the exact address. All we have to tell the operating system is which 2K boundary the character set begins at: 0, 2, 4, 6, 8, 10, 12, or 14.

And the place where we POKE the character set starting boundary is location 53272. However, we can't just POKE 53272, 14. That same byte carries other information, so we only want to affect certain bits of the byte at 53272. This is the program line we use:

```
POKE 53272, (PEEK(53272) AND 240)
OR 14
```

And there it is. Once that line is executed, the Commodore 64 looks at the block from 14K to 16K to find the character set. If your character patterns are already there, then those patterns will appear on the screen when the correct characters are PRINTed.

Changing character sets is an all-or-nothing proposition. If you change any character patterns, you have changed them all. What if you want to change only a few, and keep the rest of the letters and numbers from the character set? For instance, what if you want to put your new characters into inverse character memory, screen codes 128 through 255, and still keep the full character set from 0 to 127?

The only way to do that is to *copy* the part of character memory that you want to keep. Copying is a tricky operation, because you have to do some bank switching. And since this is a column on game programming, not high-level computer memory use, I'm not going to explain what you're doing. I'm just going to show you how to do it. This routine copies all the characters from 0 to 127 from character ROM to the 14K block.

```
400 POKE 56334,PEEK(56334)AND 254
410 POKE 1, PEEK(1) AND 251
420 CM=14:CS=1024*CM
430 FOR I=0 TO 1023:POKE CS+I,PEE
K(53248)+I:NEXT
440 POKE 1,PEEK(1) OR 4
450 POKE 56334,PEEK(56334) OR 1
```

Then, when you've copied all the non-inverse characters from the character ROM to the 14K boundary, you do this:

```
460 POKE 53272,(PEEK(53272)AND 24
0) OR 14
```

## *CHARACTER MEMORY IN THE VIC*

The VIC uses the same principle–a single POKE tells the computer where to find the character patterns. However, because the VIC is a bit more eccentric in its memory handling, there are some addi-

tional limitations, especially with the unexpanded VIC. The code for the start of character memory is POKEd into location 36869 with this command:

```
POKE 36869,PEEK(36869)AND 240 OR
codenumber
```

Character memory must start at one of the following eight addresses (each address is followed by the code for that location):

| | | |
|---|---|---|
| 32768 | 0 | (ROM characters) |
| 33792 | 1 | (ROM reversed characters) |
| 34816 | 2 | (ROM SHIFT-COMMODORE characters) |
| 35840 | 3 | (ROM reversed SHIFT-COMMODORE characters) |
| 4096 | 12 | (custom characters) |
| 5120 | 13 | (custom characters) |
| 6144 | 14 | (custom characters) |
| 7168 | 15 | (custom characters) |

Usually you will not need the entire 4K from 4096 to 8192 for your custom characters—you can only use 2K of a character set at a time, anyway. And any space you use for your character set is stolen from program space. So usually you will want to locate your character set at 6144 or 7168. If you need all 128 characters of the nonreversed character set, you must use 6144 in the unexpanded VIC. (However, if you need 64 or fewer characters, you can use 7168. In this case, though, you'll need to move screen memory, too. That's where things get complex—more complex than we have space for in this issue.)

## *USING THE EXAMPLE PROGRAM*

The example program on page 75 sets up 32 custom characters. You can use them like building blocks to make your own designs—particularly the "brick" characters from A to G. Or you can use the predesigned complex characters that are loaded into the string array CC$*(n)*.

Once you have typed in the program and RUN it, if you are using the 64 version you will see a display of custom characters and the key you must press to get each custom character. However, you must press CTRL-9 to get into reversed mode in order to get the custom characters.

With the VIC version, you'll get a blank screen—but you don't have to switch to reversed characters. (In fact, pressing CTRL-9 on a 5K or 8K VIC will show you an ever-changing character set, since the character patterns will be read from screen memory, and every character you type changes both screen memory and character memory at the same time.)

Whichever version of the program you're using, when you type the characters from A to Z and 1 to 6 you'll get pieces of drawings. You can change the colors of the figures by pressing CTRL-1 through CTRL-7 or COMMODORE-1 through COMMODORE-7. And you can switch back to the regular character set at any time by pressing CTRL-0.

To get rid of the custom character set, just press RUN-STOP/RESTORE.

In addition, if you press f1 through f7, you'll get a display of the complex characters. Ocean ships and starships, a star base, a carriage, and a castle tower give you a few samples of what can be done with combinations of custom characters.

Of course, *my* drawings aren't going to be much help for the particular needs of *your* game. To try

*Continued on page 98*

# COMMODARES

## PROGRAMMING CHALLENGES

*By Dale Rupert*

Each month, we'll present several challenges designed to toggle the bits in your cerebral random access memory. We invite you to send your solutions to:

Commodares, c/o *Ahoy!*
P.O. Box 723
Bethel, CT 06801

We will print and discuss the cleverest, simplest, shortest, most interesting and/or most unusual solutions. Be sure to identify the *name* and *number* of the problems you are solving. Also show sample runs if possible, where appropriate.

Your original programming problems would be equally welcome!

---

*Problem #12-1: Phone Fun*

Randy Barr of Liverpool, WV suggested this *Commodare.* Write a program for which you enter a phone number. The computer will print out all possible combinations of the letters on the telephone dial which equal that phone number. For example, one of the results of entering 639-4663 would be NEW-HOME, a good number for a real estate broker. We'll hold onto Randy's solution until we see some of yours. Thanks for a good idea, Randy. Be sure to send some of the best phone-words you come up with and indicate for whom they might be appropriate.

---

*Problem #12-2: More Phone Fun*

We may as well challenge you with the obvious opposite of the previous problem. The user types a seven-letter phrase, and the computer responds with the corresponding telephone number. For example, if the user types "ALL AHOY", the computer replies, "255-2469". If the user types "QUICK OX" or other invalid letters, the computer responds, "I AM SORRY, BUT WE CANNOT COMPLETE YOUR CALL AS DIALED" (or something equally clever).

---

*Problem #12-3: Letter Slalom*

Write the shortest program to slalom a letter diagonally down the screen continuously. The letter starts at the upper left corner and glides diagonally down to the right side of the screen. At that point it reverses direction and glides down to the left, always leaving a trail behind. Scrolling of the screen should make this sequence repeat indefinitely.

---

*Problem #12-4: Quick Hex*

Write the shortest program to compute the hexadecimal equivalent of any number between 0 and 65535, inclusive. For example, if the user types 16383, the computer responds with 3FFF.

---

This month we will look at readers' solutions and comments to the July and August *Commodares.* Before we begin, here is a quick tip from Greg George (Colchester, VT). He wrote in response to Michael Russell's request in the September *Commodares* for a way to make a line unlistable. Greg's solution is to include a line just before the unlistable portion of the program consisting of REM (shift-L). Now when you try to list the program, everything up to the REM is listed, but the computer stops at the (shift-L) and gives a ?SYNTAX ERROR. The (shift-L) puts a code of 204 into memory which is an invalid token value. (See how the token evaluator program in this month's *Rupert Report* avoided the problem.) Can you figure out why such a program runs properly even though it can't be listed? Thanks, Greg.

Several readers have done some serious analysis in response to *Problem #7-1: Definite Ending* which was submitted by Paul Dawson of Springhouse, PA. The problem is to write a one-line program which takes as long as possible to run. It must eventually end, and it must produce a continuously changing display (not merely scrolling). Patrick Tornudd of Helsinki, Finland submitted a program and analysis giving a duration of 2.9E+220 years. He used FOR-NEXT loops nested three deep, each running from 0 to A or −A to A in steps of 2/A where A=1.7E+38, the largest real value allowed. Unfortunately, as we discussed last month and in August, such a program will never end because of the C-64's 32-bit floating point precision. It just can't distinguish between two numbers that differ by only 1E-38 or so, even though it can represent such small values.

The situation is analogous to a ruler marked in tenths of a centimeter. It can measure something that is 100 thousand millionths of a centimeter long (0.1 cm) but it certainly can't distinguish between two objects differing by only one millionth of a centime-

ter.

This is all academic, as Patrick pointed out, since the remaining life expectancy of our sun is only 2.5E+12 years! There is one more catch that P.E. Nimmo of Fenton, Michigan brought up—the problem of energy. He claims that even if all the matter of the universe were converted to electrical energy (according to E=mc^2), the 40-watt power supply of the C-64 would exhaust that entire supply in only 2E+52 seconds. If you would like to refute or verify Mr. Nimmo's claim, be certain that you indicate all of your assumptions and calculations. We'll publish the results here.

Douglas Underwood (Walla Walla, WA) nested six FOR-NEXT loops ranging from 0 to 4.29E9 which required 8.57E48 years to finish.

P.E. Nimmo also sent the following solution to *Problem #7-1.* (You must eliminate spaces and use abbreviations to type this into one line.)

```
1 DIM C(80):FOR N=1 TO 80:J=C(N)-
I-(N=1):I=J=64:C(N)=J+I*64:PRINTC
HR$(J+32);:N=-N*(N<80-I):NEXT
```

What does this program do? It eventually prints out every possible 80-character combination of the 64 characters with ASCII codes from 32 to 96. P.E. Nimmo estimates that it will take about 1.4E+137 years to finish. He also points out that every possible one-line program will appear somewhere among the 64 to the 80th power output lines, and one of them will be the ultimate solution to *Problem #7-1.* So Mr. Nimmo may not have created the longest running program possible, but he wrote a program which *will* create it! Very impressive indeed!

Bob DeVinney (Perry, NY), John Payson (Ft. Atkinson, WV), and S.J. Mills, Jr. (Mountainside, NJ) also submitted solutions to this problem. Solutions to other July *Commodares* were submitted by Randy Barr (Liverpool, WV) and Jesus Geliga (Aquadilla, PR).

The shortest solution to *Problem #8-3: Letter Frequency* came from David Alan Wright (Middletown, CT). It is listed below.

```
1 DIM J(90):INPUT A$:FOR I=1 TO L
EN(A$):X=ASC(MID$(A$,I,1)) :J(X)=
J(X)+1:NEXT
2 FOR I=65 TO 90:IF J(I) THEN PRI
NT CHR$(I)"="J(I),
3 NEXT
```

Other readers with somewhat similar solutions include James R. Young (Grants, NM), Dave Antonison (Cambridge, ONT), Yonson Serrano (Castroville, CA), Tom Dean (Enfield, CT), Michael Velez (Lacombe, AB, Canada), and Jules Miller (Baltimore, MD).

Jim Brice (Anchorage, AK) used a GET statement as part of the main loop, so the user is not limited to typing only 80 characters. His program is listed below.

```
10 DIM N(255)
20 PRINT "TYPE A SENTENCE AND <RE
TURN>"
30 GET A$ :IF A$="" THEN 30
40 PRINT A$;
50 IF A$=CHR$(13) THEN 80
60 P=ASC(A$):N(P)=N(P)+1
70 GOTO 30
80 FOR I=32 TO 255
90 IF N(I)=0 THEN 110
100 PRINT CHR$(I)"="N(I),
110 NEXT I
```

Many others submitted solutions to this problem and to *Problem #8-2: Bug Eliminator,* including Mike Brumit (Deer Park, TX), Doug Mataconia (Piscataway, NJ), Malcolm Davidson (Ransomville, NY), Martin Levinton (New Rochelle, NY), Adam Fonti (Hawthorne, NY), J.L. Tyler (Rochester, IN), Roy Martin (Salinas, CA), John Daugherty (Billings, MT), David Frazier (Hudson, OH), Marcel J. Devoe (Peabody, MA), Bill Feyler (Pleasant Plain, OH), and Karl Steger (St. Louis, MO).

Thanks also to Baertschi Christian of Ascona, Switzerland for the nice letter and solutions to the August *Commodares. Ahoy!* is far-reaching!

Most *Bug Eliminator* solutions sequentially PEEKed into screen memory until the bug was found. Even though I tried to fix the bug in the problem last month, Michael Velez pointed out that the value of L is just one more than the position of the bug. You may eliminate the bug by POKEing into the L-1 screen location even after homing the cursor.

Not many people sent useful functions in response to *Problem #8-4: Useful Functions.* David Alan Wright (Middletown, CT) suggested the following function to return the number of minutes since TI$ was set:

```
1 DEF FNMIN(X)=VAL(TI$)/360
```

He sent similar ones for seconds and hours.

Sol Steinberg (Bellmawr, NJ) commented that it is possible to use functions on the right side of the definition of another function, for example:

```
1 DEF FNA(X)=X*3.14159/180
2 DEF FNB(X)=SIN(FNA(X))
```

The function FNB(X) will return the value of the sine of X where X is in degrees (rather than radi-

*Continued on page 98*




# Big Value, Small Price!

Why is quality software so expensive?

No wonder people keep asking that question — some of the most popular business packages cost $50, $100, $200, or more!

Well, at DATAMOST we not only asked the same question, we did something about solving the problem of over-priced software. We created KWIK-WARE!™ at only $19.95 per package.

KWIK-WARE! is an exciting new line of Commodore 64 software that includes KWIK-LOAD!™ and KWIK-WRITE!™, as well as other soon-to-be-released software products.

KWIK-LOAD! is a menu-driven, disk utility program for the Commodore 1541 disk drive that allows you to load programs, save files, and copy disks over three times faster than with normal Commodore DOS! KWIK-LOAD! also lets you test the operating speed of your drive, edit information on disk by track or sector, rename files, delete files, validate and format disks, print out sectors, and display and print disk directories, FAST! KWIK-LOAD! helps you save time, which is one of the big reasons you bought your Commodore 64.

And it costs only $19.95!




KWIK-WRITE! is a full-featured word processor for the Commodore 64 with all the text-editing functions of more costly word processors. It includes easy-to-use command menus and help screens, cut and paste, search and replace, print preview, underlining, screen displays up to 132 characters wide, and more.

KWIK-WRITE! supports most popular printers, loads and saves files fast with the help of built-in KWIK-LOAD!, includes a KWIK-REFERENCE! card, and is compatible with other KWIK-WARE! products.

Priced at just $19.95, KWIK-WRITE! delivers the quality and power of the most popular word processors, at a fraction of the price!

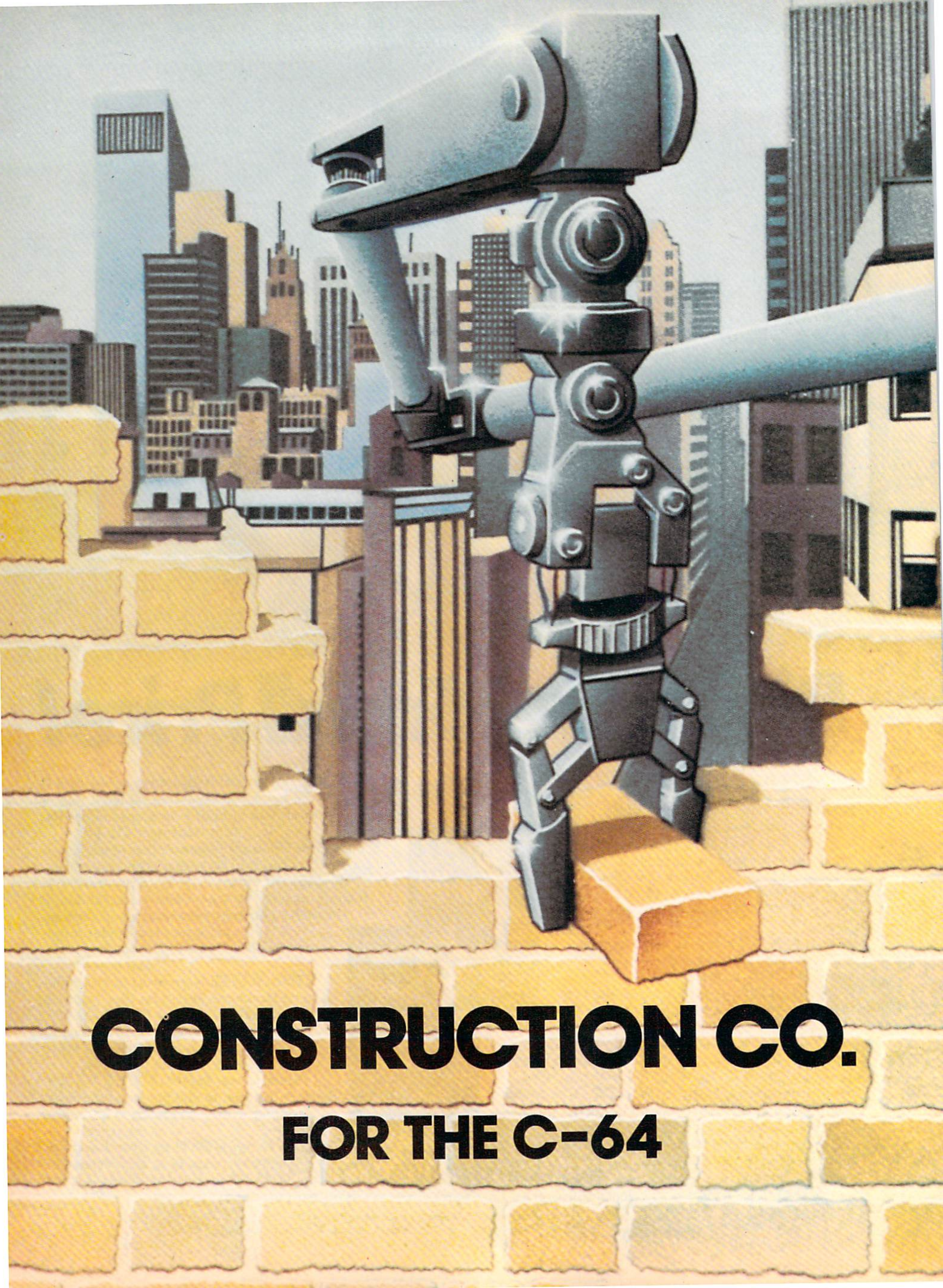
**At DATAMOST, we didn't just ask why software is so expensive. We did something about it.**

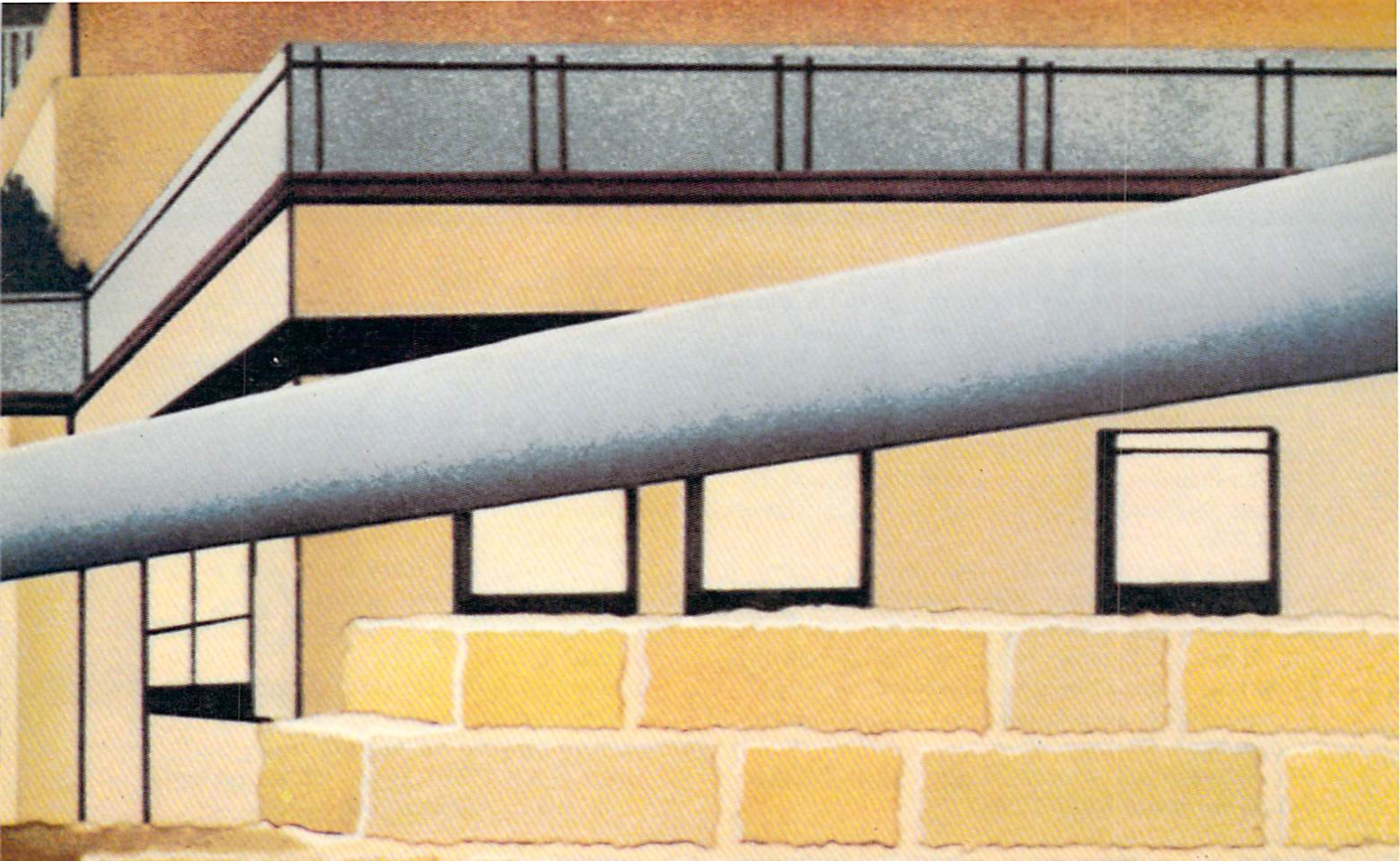
DATAMOST™ INC

20660 Nordhoff St., Chatsworth, CA 91311 (818) 709-1202

Commodore 64 and 1541 are registered trademarks of Commodore Business Machines, Inc.
KWIK-LOAD!, KWIK-WRITE!, and KWIK-WARE! are registered trademarks of DATAMOST, Inc.

Reader Service No. 44


CONSTRUCTION CO.
FOR THE C-64



BY B.W. BEHLING

It's a nice summer day and you decide that a backyard barbecue would be just the thing to put the family in a good mood. Later, while the hot dogs and hamburgers are slowly cooking, you have a chat with your neighbor, the real estate broker. While he's mowing his lawn he tells you about some property he has that he wants to build on. If he could just find a contractor he could trust he would be able to put up some houses and sell them to his customers at a nice profit. "You wouldn't know a good contractor?" he asks. You think quickly. You've been out of work since March, when the butter churn factory closed, and are getting low on cash. You then tell your neighbor that you would be glad to build those houses for him if you can agree on a fair price. He offers you $50,000 for every house you can build and you meet in his office the next day to sign the papers.

Contract in hand, your next stop is the bank where you take a $35,000 loan at 12% interest with your own home as security. Unfortunately, the loan is only for 30 days, but you think you can have the first house finished and sold by that time. Now the only problem is the fact that the only house you ever built was a birdhouse, and even the birds won't touch it. Then you realize that you don't have to do the actual work yourself if you just hire experienced laborers.

Offering $70 a day, you make the rounds of the local bars looking for anyone who knows which end of a nail you hit with a hammer. Soon you have enough people to build a small city. You explain what you're building and give a rousing speech (aided by several stiff shots and a couple dozen beers) and tell everyone to show up for work in the morning.

The next morning when you get to the job site you find about 50 people waiting to work. You quickly come to the conclusion that you can only keep an eye on a maximum of 9 at a time. You tell the rest to come back tomorrow, promising to hire new people every day. Taking your truck, you head to the local building supply to get your materials.

Not knowing how much of anything to

© John P. Tierney 1984

buy you consult the plans for the house, but all you can understand is that you'll need about 25 windows, 20 doors, and a garage door for each house.

After buying what you hope will be enough materials you chillingly remember that you told your workers that if at any time they ran out of materials they would be paid for the full day and sent home. With thoughts of living in a cave for the next 20 years you head for the job and (you hope) success.

NOTES:

Since you can only hire one type of worker at a time (i.e., carpenters), when one type of job is finished everyone goes home for the day.

If you are fortunate enough to pay off your loan on time, you will be given the chance to borrow more money. Keep in mind that all loans are for 30 days only.

To make your planning easier I'll tell you now how the house will be built.

| JOB | TYPE OF WORKER USED |
|---|---|
| 1. Foundation | Bricklayer |
| 2. Frame | Carpenter |
| 3. Roof | Roofer |
| 4. Plumbing & Electric | Utilityman |
| 5. Sides of House | Carpenter |
| 6. Plaster | Plasterer |
| 7. Windows | Glazier |
| 8. Fence, Doors and Garage Door | Carpenter |
| 9. Sidewalk | Mason |
| 10. Painting | Painter |

## PLAYING TIPS

1. Materials can be stolen overnight so try not to stockpile materials unless you can find a really good price.
2. Plumbing and wiring takes more than 150 man hours to complete.
3. A good bricklayer can do 100 bricks per hour.
4. Plastering takes more than 375 man hours to complete.
5. Try to judge how many hours work are left on a job and hire workers accordingly.
6. Always try to buy materials for the lowest price possible.
7. Keeping track of materials used on your first house will make later houses easier.
8. The record for building one house is 23 days at a cost of $29,452.63. If you top that, drop me a line and let me know how you did it.

Special thanks to my wife Marie who suggested the idea for this game, and did all the typing without getting a chance to play the game before it was sold. Love you honey.

For a copy of this or any of my programs on *cassette only,* send a blank tape and a self-addressed stamped mailer with $5.00 and the program name to:

B.W. Behling
232 Jackson St.
Brooklyn, N.Y. 11211

Or order the *Ahoy!* program disk, containing all the programs in this issue. See page 66 for details.

*SEE PROGRAM LISTING ON PAGE 64*

**As of last issue we began utilizing an improved method of printing our program listings. This new method will make entering programs easier for you, but only if you read this entire page carefully before proceeding.**

On the following pages you'll find several programs that you can enter on your Commodore computer. But before doing so, read this entire page carefully.

To insure clear reproductions, *Ahoy!*'s program listings are generated on a daisy wheel printer, incapable of printing the commands and graphic characters used in Commodore programs. These are therefore represented by various codes enclosed in brackets [ ]. For example: the SHIFT CLR/HOME command is represented onscreen by a heart. The code we use in our listings is [CLEAR]. The chart below lists all such codes which you'll encounter in our listings, except for one other special case.

The other special case is the COMMODORE and SHIFT characters. On the front of most keys are two symbols. The symbol on the left is obtained by pressing that key while holding down the COMMODORE key; the symbol on the right, by pressing that key while holding down the SHIFT key. COMMODORE and SHIFT characters are represented in our listings by a lower-case "s" or "c" followed by the symbol of the key you must hit. COMMODORE J, for example, is represented by [c J], and SHIFT J by [s J].

Additionally, any character that occurs more than two times in a row will be displayed by a coded listing. For example, [3 "[LEFT]"] would be 3 CuRSoR left commands in a row, [5 "[s EP]"] would be 5 SHIFTed English Pounds, and so on. Multiple blank spaces will be noted in similar fashion: 22 spaces, for example, as [22 " "].

Sometimes you'll find a program line that's too long for the computer to accept (C-64 lines are a maximum of 80 characters, or 2 screen lines, long; VIC 20 lines, a maximum of 88 characters, or 4 screen lines). To enter these lines, refer to the *BASIC Command Abbreviations Appendix* in your User Manual.

On the next page you'll find our *Bug Repellent* programs for the VIC 20 and C-64. The version appropriate for your machine will help you proofread our programs after you type them. (Please note: the *Bug Repellent Line Codes* that follow each program listing should *not* be typed in. See the instructions preceding each program.) □

| When You See | It Means | You Type | You Will See |
|---|---|---|---|
| [CLEAR] | Screen Clear | SHIFT CLR/HOME | |
| [HOME] | Home | CLR/HOME | |
| [UP] | Cursor Up | SHIFT ↑ CRSR ↓ | |
| [DOWN] | Cursor Down | ↑ CRSR ↓ | |
| [LEFT] | Cursor Left | SHIFT ←CRSR→ | |
| [RIGHT] | Cursor Right | ←CRSR→ | |
| [SS] | Shifted Space | SHIFT Space | |
| [INSERT] | Insert | SHIFT INST/DEL | |
| [DEL] | Delete | INST/DEL | |
| [RVSON] | Reverse On | CNTRL 9 | |
| [RVSOFF] | Reverse Off | CNTRL 0 | |
| [UPARROW] | Up Arrow | ↑ | |
| [BACKARROW] | Back Arrow | ← | |
| [PI] | PI | π | |
| [EP] | English Pound | £ | |

| When You See | It Means | You Type | You Will See |
|---|---|---|---|
| [BLACK] | Black | CNTRL 1 | |
| [WHITE] | White | CNTRL 2 | |
| [RED] | Red | CNTRL 3 | |
| [CYAN] | Cyan | CNTRL 4 | |
| [PURPLE] | Purple | CNTRL 5 | |
| [GREEN] | Green | CNTRL 6 | |
| [BLUE] | Blue | CNTRL 7 | |
| [YELLOW] | Yellow | CNTRL 8 | |
| [F1] | Function 1 | F1 | |
| [F2] | Function 2 | SHIFT F1 | |
| [F3] | Function 3 | F3 | |
| [F4] | Function 4 | SHIFT F3 | |
| [F5] | Function 5 | F5 | |
| [F6] | Function 6 | SHIFT F5 | |
| [F7] | Function 7 | F7 | |
| [F8] | Function 8 | SHIFT F7 | |

**IMPORTANT!**
**Before typing in the *Bug Repellent* and other *Ahoy!* programs, refer to the information on page 61.**

# VIC 20 BUG REPELLENT

## By Michael Kleinert and David Barron

The program listed below will allow you to quickly debug any *Ahoy!* program you type in on your VIC 20. Follow directions for cassette or disk.

*For cassette:* type in and save the *Bug Repellent* program, then type RUN 63000[RETURN]SYS 828[RETURN]. If you typed the program properly, it will generate a set of two-letter line codes that will match those listed below the program on this page. (If you didn't type the program properly, of course, no line codes will be generated. You'll have to debug the *Bug Repellent* itself the hard way.)

Once you've got a working *Bug Repellent,* type in the program you wish to check. Save it and type the RUN and SYS commands listed above once again, then compare the line codes generated to those listed in the magazine. If you spot a discrepancy, a typing error exists in that line. Important: you must use exactly the same spacing as the program in the magazine. Due to memory limitations on the VIC, the VIC *Bug Repellent* will register an error if your spacing varies from what's printed.

You may type SYS 828 as many times as you wish, but if you use the cassette for anything, type RUN 63000 to restore the *Repellent.*

When your program has been disinfected you may delete all lines from 63000 on. (Be sure the program you type doesn't include lines above 63000!)

*For disk:* type in the *Bug Repellent,* save it, and type RUN:NEW[RETURN]. (See above regarding testing the *Bug Repellent* on itself.) Type in the program you wish to check, then SYS 828. This will generate a set of two-letter line codes that you should compare to those listed in the magazine.

To pause the line codes listing, press SHIFT. To permanently pause it, press SHIFT LOCK. To continue, release SHIFT LOCK.

To send the list to the printer type OPEN 4,4:CMD 4:SYS 828[RETURN]. When the cursor comes back, type PRINT#4:CLOSE 4[RETURN].

```
•63000 FOR X = 828 TO 1023 :READ Y
:POKE X,Y:NEXT:END
•63001 DATA 169, 0, 133, 63, 133,
64, 165, 43, 133, 251
•63002 DATA 165, 44, 133, 252, 160
, 0, 132, 254, 32, 228
•63003 DATA 3, 234, 177, 251, 208,
 3, 76, 208, 3, 230
•63004 DATA 251, 208, 2, 230, 252,
 169, 244, 160, 3, 32
•63005 DATA 30, 203, 160, 0, 177,
251, 170, 230, 251, 208
•63006 DATA 2, 230, 252, 177, 251,
 32, 205, 221, 169, 58
•63007 DATA 32, 210, 255, 169, 0,
133, 253, 230, 254, 32
•63008 DATA 228, 3, 234, 165, 253,
 160, 0, 170, 177, 251
•63009 DATA 201, 32, 240, 6, 138,
113, 251, 69, 254, 170
•63010 DATA 138, 133, 253, 177, 25
1, 208, 226, 165, 253, 41
•63011 DATA 240, 74, 74, 74, 74, 2
4, 105, 65, 32, 210
•63012 DATA 255, 165, 253, 41, 15,
 24, 105, 65, 32, 210
•63013 DATA 255, 169, 13, 32, 210,
 255, 173, 141, 2, 41
•63014 DATA 1, 208, 249, 230, 63,
208, 2, 230, 64, 230
•63015 DATA 251, 208, 2, 230, 252,
 76, 74, 3, 169, 236
•63016 DATA 160, 3, 32, 30, 203, 1
66, 63, 165, 64, 32
•63017 DATA 205, 221, 169, 13, 32,
 210, 255, 96, 230, 251
•63018 DATA 208, 2, 230, 252, 96,
0, 76, 73, 78, 69
•63019 DATA 83, 58, 32, 0, 76, 73,
 78, 69, 32, 35
•63020 DATA 32, 0, 0, 0, 0, 0
```

### BUG REPELLENT LINE CODES FOR VIC 20 BUG REPELLENT

```
LINE # 63000:MH    LINE # 63011:NN
LINE # 63001:BD    LINE # 63012:IG
LINE # 63002:FO    LINE # 63013:EN
LINE # 63003:ND    LINE # 63014:GJ
LINE # 63004:DJ    LINE # 63015:IK
LINE # 63005:LP    LINE # 63016:HG
LINE # 63006:JB    LINE # 63017:CK
LINE # 63007:JF    LINE # 63018:JF
LINE # 63008:KA    LINE # 63019:OH
LINE # 63009:HP    LINE # 63020:LH
LINE # 63010:KJ    LINES: 21
```

# C-64 BUG REPELLENT

## By Michael Kleinert and David Barron

The program listed below will allow you to quickly debug any *Ahoy!* program you type in on your C-64.

Type in, SAVE, and RUN the *Bug Repellent*, Type NEW, then type in or LOAD the *Ahoy!* program you wish to check. When that's done, SAVE your program (don't RUN it!) and type SYS 49152 [RETURN]. You'll be asked if you want the line value codes displayed on the screen or dumped to the printer. If you select screen, it will appear there.

The table will move quickly, too quickly for most mortals to follow. To pause the listing depress and hold the SHIFT key. To pause for an extended period, depress SHIFT LOCK. As long as it is locked, the display will remain frozen.

Compare the table your machine generates to the table in *Ahoy!* that follows the program you're entering: If you spot a difference, an error exists in that line. Jot down the numbers of lines where contradictions occur, LIST each line, spot the errors, and correct them.

```
•5000 FOR X = 49152 TO 49488 :READ
 Y:POKE X,Y:NEXT:END
•5001 DATA 32, 161, 192, 165, 43,
133, 251, 165, 44, 133
•5002 DATA 252, 160, 0, 132, 254,
32, 37, 193, 234, 177
•5003 DATA 251, 208, 3, 76, 138, 1
92, 230, 251, 208, 2
•5004 DATA 230, 252, 76, 43, 192,
76, 73, 78, 69, 32
•5005 DATA 35, 32, 0, 169, 35, 160
, 192, 32, 30, 171
•5006 DATA 160, 0, 177, 251, 170,
230, 251, 208, 2, 230
•5007 DATA 252, 177, 251, 32, 205,
 189, 169, 58, 32, 210
•5008 DATA 255, 169, 0, 133, 253,
230, 254, 32, 37, 193
•5009 DATA 234, 165, 253, 160, 0,
76, 13, 193, 133, 253
•5010 DATA 177, 251, 208, 237, 165
, 253, 41, 240, 74, 74
•5011 DATA 74, 74, 24, 105, 65, 32
, 210, 255, 165, 253
•5012 DATA 41, 15, 24, 105, 65, 32
, 210, 255, 169, 13
•5013 DATA 32, 220, 192, 230, 63,
208, 2, 230, 64, 230
•5014 DATA 251, 208, 2, 230, 252,
76, 11, 192, 169, 153
•5015 DATA 160, 192, 32, 30, 171,
166, 63, 165, 64, 76
•5016 DATA 231, 192, 96, 76, 73, 7
8, 69, 83, 58, 32
•5017 DATA 0, 169, 247, 160, 192,
32, 30, 171, 169, 3
•5018 DATA 133, 254, 32, 228, 255,
 201, 83, 240, 6, 201
•5019 DATA 80, 208, 245, 230, 254,
 32, 210, 255, 169, 4
•5020 DATA 166, 254, 160, 255, 32,
 186, 255, 169, 0, 133
•5021 DATA 63, 133, 64, 133, 2, 32
, 189, 255, 32, 192
•5022 DATA 255, 166, 254, 32, 201,
 255, 76, 73, 193, 96
•5023 DATA 32, 210, 255, 173, 141,
 2, 41, 1, 208, 249
•5024 DATA 96, 32, 205, 189, 169,
13, 32, 210, 255, 32
•5025 DATA 204, 255, 169, 4, 76, 1
95, 255, 147, 83, 67
•5026 DATA 82, 69, 69, 78, 32, 79,
 82, 32, 80, 82
•5027 DATA 73, 78, 84, 69, 82, 32,
 63, 32, 0, 76
•5028 DATA 44, 193, 234, 177, 251,
 201, 32, 240, 6, 138
•5029 DATA 113, 251, 69, 254, 170,
 138, 76, 88, 192, 0
•5030 DATA 0, 0, 0, 230, 251, 208,
 2, 230, 252, 96
•5031 DATA 170, 177, 251, 201, 34,
 208, 6, 165, 2, 73
•5032 DATA 255, 133, 2, 165, 2, 20
8, 218, 177, 251, 201
•5033 DATA 32, 208, 212, 198, 254,
 76, 29, 193, 0, 169
•5034 DATA 13, 76, 210, 255, 0, 0,
 0
```

## BUG REPELLENT LINE CODES FOR C-64 BUG REPELLENT

```
LINE # 5000:GJ       LINE # 5018:FK
LINE # 5001:DL       LINE # 5019:FL
LINE # 5002:DB       LINE # 5020:CL
LINE # 5003:OF       LINE # 5021:GC
LINE # 5004:KN       LINE # 5022:NN
LINE # 5005:CA       LINE # 5023:NH
LINE # 5006:CE       LINE # 5024:IM
LINE # 5007:JE       LINE # 5025:KC
LINE # 5008:CL       LINE # 5026:DC
LINE # 5009:NB       LINE # 5027:ML
LINE # 5010:MB       LINE # 5028:GN
LINE # 5011:EP       LINE # 5029:JK
LINE # 5012:GH       LINE # 5030:NA
LINE # 5013:AN       LINE # 5031:DM
LINE # 5014:NG       LINE # 5032:JA
LINE # 5015:BF       LINE # 5033:FM
LINE # 5016:EP       LINE # 5034:PA
LINE # 5017:PJ       LINES: 35
```

# CONSTRUCTION CO.

FROM PAGE 60

```
•2 POKE53280,6:POKE53281,1:PRINT"
[CLEAR][BLACK][4"[DOWN]"]";TAB(6
);"[28"#"]"
•4 PRINTTAB(6);"[4"#"[21" "[5"#"]
"
•6 PRINTTAB(6);"[4"#"[3" "]CONSTR
UCTION CO.  [4"#"]"
•8 PRINTTAB(6);"[4"#"[21" "[5"#"]
"
•10 PRINTTAB(6);"[4"#"[4" "]BY B.
W.BEHLING[3" "[5"#"]"
•12 PRINTTAB(6);"[4"#"[21" "[5"#"
]"
•14 PRINTTAB(6);"[4"#"] (C) COPYR
IGHT 1984 [4"#"]"
•16 PRINTTAB(6);"[4"#"[21" "[5"#"
]"
•18 PRINTTAB(6);"[28"#"]"
•20 PRINTTAB(9);"[7"[DOWN]"][RVSO
N][BLUE] HIT ANY KEY TO START [B
LACK]"
•22 GETK$:IFK$=""THEN22
•24 PRINT"[UP] [RVSON][GREEN]HOW
MANY HOUSES WILL YOU BUILD?  (1-
5)[WHITE]"
•26 GETK$:IFK$=""THEN26
•28 IFASC(K$)<49ORASC(K$)>53THEN2
6
•30 HN=VAL(K$):PRINT"[CLEAR]"
•32 DIMMA(11,3),MA$(11,2):FORI=1T
O11:MA(I,3)=0:NEXTI
•34 SC=1024:CL=55296:WP=70:DA=0:H
O=1:CA=35000:LA=35000+(35000*.12
):LD=30
•36 FORI=1TO11:READMA(I,1):READMA
(I,2):NEXTI
•38 FORA=1TO11:FORB=0TO2:READMA$(
A,B):NEXTB:NEXTA
•40 PRINT"[CLEAR]":POKE53280,4:PO
KE53281,0:GOSUB100:GOSUB510:GOTO
800
•100 PRINT"[HOME]";TAB(27);"[WHIT
E][RVSON]DAY  HOURS":PRINT"[DOWN
]";TAB(27);DA;TAB(33);HC
•102 PRINTTAB(27);"[RVSON][DOWN]$
";CA:PRINTTAB(27);"[DOWN]WORKERS
";WO:PRINTTAB(27);"[DOWN][RVSON]
MATERIALS"
•104 PRINT:FORI=1TO11:PRINTTAB(27
);MA$(I,0);MA(I,3):NEXTI:RETURN
•150 TR=TR-249:ONTRGOTO160
•152 IFDA=1THENRETURN
•154 MSG$=" YOUR MATERIALS WERE S
TOLEN OVERNIGHT!!":GOSUB510
•156 GOSUB500:PRINT"[HOME][9"[DOW
N]"]":FORS=1TO11:PRINTTAB(33);"[
6" "]"
•158 MA(S,3)=0:NEXTS:GOSUB100:GOS
UB190:GOTO604
•160 MSG$="[UP] YOU WERE ARRESTED
 FOR BRIBING A ":GOSUB510:GOSUB5
00
•164 MSG$=" BUILDING INSPECTOR. N
O WORK TODAY!":GOSUB500:GOSUB190
:GOTO600
•190 FORD=1TO2000:NEXTD:RETURN
•200 DEFFNRN(VA)=(.01)*INT(RND(0)
*VA)
•202 FORI=1TO11:MA(I,0)=MA(I,1)+F
NRN(MA(I,2)):NEXTI:FORP=1TO11
•204 PRINT"[HOME][20"[DOWN]"]"
•206 IFMA(P,0)<MA(P,1)+((MA(P,2)*
.01)/2)THEN210
•208 PRINT"[WHITE][RVSON]I HAVE "
;MA$(P,1);" @ $";MA(P,0);"[LEFT]
 ";MA$(P,2):GOTO212
•210 PRINT"[CYAN][RVSON]I HAVE ";
MA$(P,1);" ON SALE @ $";MA(P,0);
"[LEFT] ";MA$(P,2)
•212 INPUT"[WHITE][RVSON]HOW MANY
 DO YOU WANT";QU
•214 IFQU<0THENP=11:GOTO228
•216 IFQU+MA(P,3)<=99999THEN220
•218 MSG$=" YOU DON'T HAVE ROOM F
OR THAT MUCH!! ":GOSUB510:GOSUB5
00:GOTO224
•220 IFQU*MA(P,0)<=CATHEN226
•222 MSG$=" YOU CAN'T AFFORD THAT
! ":GOSUB510:GOSUB500
•224 GOSUB190:GOSUB510:GOTO204
•226 MA(P,3)=MA(P,3)+QU:PP=MA(P,0
)*QU:CA=CA-PP:CA=.01*INT(CA*100)
•228 GOSUB350:GOSUB100:GOSUB510:N
EXTP:RETURN
•250 PRINT"[HOME][8"[DOWN]"]":FOR
E=1TOM:PRINT:NEXTE
•252 PRINTTAB(33);"[6" "]":GOSUB1
00:IFM=1THENM=2:GOTO250
•254 HD=INT(HR/WO):IFHD>HCTHENHC=
HD:GOSUB100
•256 IFHR<QTTHENRETURN
```

**IMPORTANT**
Before typing in an *Ahoy!* program, refer to the first two pages of the program listings section.

```
•258 HR=0:CA=CA-(70*WO):MSG$=" WO
RK FINISHED FOR TODAY. CREW PAID
.":GOSUB500
•260 GOSUB350:GOSUB190:HC=0:GOSUB
600:RETURN
•300 MSG$=" HOW MUCH? 1=$15[3"0"]
 2=$3[4"0"] 3=$45[3"0"] ":GOSUB5
00:LD=DA+30
•302 GETK$:IFK$=""THEN302
•304 IFASC(K$)<49ORASC(K$)>51THEN
302
•306 LA=15000*VAL(K$):CA=CA+LA:LA
=LA+(LA*.12):GOSUB510
•350 CA=.01*INT(CA*100)
•352 PRINTCHR$(19);"[HOME][4"[DOW
N]"][WHITE]";TAB(27);"[12" "]";"
[12"[LEFT]"][RVSON]$";CA:RETURN
•400 MSG$=" YOU ARE BANKRUPT[4"!"
] ":GOSUB500:GOSUB190
•402 PRINT"[CLEAR][4"[DOWN]"][BLA
CK]":POKE53281,1:GOTO936
•500 PRINT"[HOME][WHITE][RVSON][2
2"[DOWN]"]";MSG$:RETURN
•510 FORI=840TO1000:POKESC+I,160:
POKECL+I,5:NEXTI:RETURN
•600 DA=DA+1:PRINT"[HOME][DOWN][D
OWN]";TAB(27);DA;TAB(33);HC
•602 TR=INT(RND(0)*250)+1:IFTR>24
8THENGOTO150
•604 GOSUB510:IFDA<>LDTHEN610
•606 MSG$=" YOUR LOAN IS DUE TODA
Y.":GOSUB500:GOSUB190:GOSUB510
•608 CA=CA-LA:LA=0:LD=0:GOSUB350
•610 IFCA<=0THENGOTO400
•612 IFLA<>0 OR CA>4999THEN620
•614 MSG$=" DO YOU WANT TO TAKE A
 LOAN? ":GOSUB510:GOSUB500
•616 GETK$:IFK$=""THEN616
•618 IFK$="Y"THENGOSUB510:GOSUB30
0
•620 GOSUB510:MSG$=" HOW MANY WOR
KERS DO YOU NEED TODAY? ":GOSUB5
00
•622 MSG$="[DOWN] ENTER NUMBER. (
1-9) [HOME]":GOSUB500
•624 GETK$:IFK$=""THEN624
•626 IFASC(K$)<49ORASC(K$)>57THEN
624
•628 WO=VAL(K$):QT=WO*8:PRINT"[HO
ME][6"[DOWN]"][WHITE]";TAB(34);W
O:GOSUB510
•630 MSG$=" DO YOU NEED MATERIALS
? ":GOSUB500
•632 GETK$:IFK$=""THEN632
•634 IFK$="Y"THENGOSUB510:GOSUB20
0
•636 GOSUB510:MSG$=" WORK IN PROG
RESS. ":GOSUB500:RETURN
•700 GOSUB250:PRINT"[HOME][22"[DO
WN]"][WHITE][RVSON] YOU RAN OUT
OF MATERIALS!!"
•702 PRINT"[RVSON] CREW WAS PAID
AND SENT HOME AFTER";HC;"[LEFT]
HRS[HOME]":CA=CA-(70*WO):HC=0
•704 GOSUB350:HR=0:GOSUB190:GOSUB
600:RETURN
•800 GOSUB600:BU=0:HR=0
•802 BU=BU+10:MA(1,3)=MA(1,3)-10:
MA(2,3)=MA(2,3)-8:HR=HR+.1:M=1
•804 IFMA(1,3)<0 OR MA(2,3)<0THEN
GOSUB700
•806 IFBU<100THENGOSUB250:GOTO802
•808 BU=0:READX:READZ:IFX=-1THENB
U=0:HR=QT:GOTO812
•810 POKESC+X,Z+200:POKECL+X,9:GO
SUB250:GOTO802
•812 MSG$=" BRICK FINISHED!":GOSU
B510:GOSUB500:GOSUB190:M=3:GOTO8
16
•814 LU=LU+3:MA(3,3)=MA(3,3)-3:IF
MA(3,3)<0THENGOSUB700
•816 GOSUB250:IFLU=6THENLU=0:HR=H
R+1.5
•818 READX:READZ:IFX=-1THENLU=0:H
R=QT:GOTO822
•820 POKESC+X,Z:POKECL+X,7:GOTO81
4
•822 MSG$=" FRAME FINISHED!":GOSU
B510:GOSUB500:GOSUB190:M=4:GOTO8
26
•824 SH=SH+2:MA(4,3)=MA(4,3)-2:IF
MA(4,3)<0THENGOSUB700
•826 GOSUB250:IFSH<10THEN824
•828 SH=0:HR=HR+1:READX:READZ:IFX
=-1THENSH=0:HR=QT:GOTO832
•830 POKESC+X,Z:POKECL+X,11:GOTO8
24
•832 MSG$=" ROOF FINISHED!":GOSUB
510:GOSUB500:GOSUB190:M=5:GOTO83
6
•834 M=5:PI=PI+3:MA(5,3)=MA(5,3)-
3:IFMA(5,3)<0THENGOSUB700
•836 GOSUB250
•838 M=6:MA(6,3)=MA(6,3)-7:IFMA(6
,3)<0THENGOSUB700
•840 GOSUB250:HR=HR+1:IFPI<499THE
```

## IMPORTANT

Before typing in an *Ahoy!* program, refer to the first two pages of the program listings section.

```
N834
•842 PI=0:HR=QT:MSG$=" PLUMBING &
 WIRING FINISHED!":GOSUB510:GOSU
B500
•844 GOSUB190:M=3:GOTO848
•846 LU=LU+3:MA(3,3)=MA(3,3)-3:IF
MA(3,3)<0THENGOSUB700
•848 GOSUB250:IFLU=6THENLU=0:HR=H
R+1
•850 READX:IFX=-1THEN858
•852 POKESC+X,160:POKECL+X,7:GOTO
846
•854 LU=LU+3:MA(3,3)=MA(3,3)-3:IF
MA(3,3)<0THENGOSUB700
•856 GOSUB250:IFLU=6THENLU=0:HR=H
R+1
•858 READX:READZ:IFX=-1THENLU=0:H
R=QT:GOTO862
•860 POKESC+X,Z:POKECL+X,7:GOTO85
4
•862 MSG$=" SIDES FINISHED!":GOSU
B510:GOSUB500:GOSUB190:M=7:GOTO8
66
•864 PL=PL+3:MA(7,3)=MA(7,3)-3:IF
MA(7,3)<0THENGOSUB700
•866 GOSUB250:HR=HR+.5:IFPL<799TH
EN864
•868 MSG$=" PLASTERING FINISHED!"
:GOSUB510:GOSUB500:PL=0:HR=QT
•870 GOSUB190:M=8:GOTO874
•872 HR=HR+1.5:MA(8,3)=MA(8,3)-1:
IFMA(8,3)<0THENGOSUB700
•874 GOSUB250
•876 FORT=1TO3:READX:READZ:IFX=-1
THENT=0:HR=QT:GOTO880
•878 POKESC+X,Z:POKECL+X,1:NEXTT:
GOTO872
•880 MSG$=" WINDOWS FINISHED!":GO
SUB510:GOSUB500:GOSUB190:M=3:GOT
O884
•882 HR=HR+.5:MA(3,3)=MA(3,3)-4:I
FMA(3,3)<0THENGOSUB700
•884 GOSUB250:READX:IFX=-1THEN888
•886 POKESC+X,35:POKECL+X,1:GOTO8
82
•888 MSG$=" FENCE FINISHED!":GOSU
B510:GOSUB500:GOSUB190:M=9:GOSUB
636
•890 HR=HR+2:MA(9,3)=MA(9,3)-1:IF
MA(9,3)<0THENGOSUB700
•892 GOSUB250:READX:READZ:IFX=-1T
HEN896
•894 POKESC+X,Z:POKECL+X,8:GOTO89
0
•896 M=10:GOSUB636
•898 IFMA(10,3)<=0THENGOSUB700
•900 HR=HR+.25:GOSUB250:READX:REA
DZ:IFX=-1THEN904
•902 POKESC+X,Z:POKECL+X,8:GOTO90
0
•904 MA(10,3)=MA(10,3)-1:GOSUB250
:MSG$=" DOORS FINISHED!":GOSUB51
0
•906 GOSUB500:GOSUB190:M=2:HR=QT:
GOTO910
•908 HR=HR+1.5:MA(2,3)=MA(2,3)-10
:IFMA(2,3)<0THENGOSUB700
•910 GOSUB250:READX:IFX=-1THENHR=
QT:GOTO914
•912 X=X+810:POKESC+X,160:POKECL+
X,12:GOTO908
•914 MSG$=" SIDEWALK FINISHED!":G
OSUB510:GOSUB500:GOSUB190:GOSUB2
50:M=11:CO=13
•916 PA=PA+1:IFMA(11,3)<0THENGOSU
B700
•918 READX:IFX=-1THENCO=CO+1:IFCO
=15THEN924
•920 POKECL+X,CO:IFPA<5THEN916
•922 HR=HR+4:MA(11,3)=MA(11,3)-1:
GOSUB250:PA=0:GOTO916
•924 HO=HO+1:CA=CA+50000:GOSUB350
:IFHO>HNTHEN934
•926 MSG$="[UP] HOUSE FINISHED!!"
:GOSUB510:GOSUB500
•928 MSG$=" HIT ANY KEY TO START
NEXT HOUSE.":GOSUB500
•930 GETK$:IFK$=""THEN930
•932 RESTORE:GOTO36
•934 PRINT"[CLEAR]":POKE53281,1:P
RINT"[6"[DOWN]"]";TAB(6);"[BLACK
]YOU FINISHED THE CONTRACT!!"
•936 CA=CA-LA:CA$=STR$(CA):IFVAL(
CA$)<0THEN944
•938 PRINTTAB(4);"[DOWN][DOWN]AFT
ER PAYING YOUR DEBTS YOU MADE:"
•940 PRINTTAB(19-INT(LEN(CA$)/2))
;"[DOWN][DOWN]$";CA$
•942 PRINTTAB(11);"[DOWN][DOWN]CO
NGRATULATIONS[3"!"]":GOTO950
•944 PRINTTAB(7);"[DOWN][DOWN]TOO
 BAD YOU OWE THE BANK:"
•946 PRINTTAB(19-INT(LEN(CA$)/2))
;"[DOWN][DOWN]$";CA$
•948 PRINTTAB(5);"[DOWN][DOWN]AND
 HAD TO RETIRE IN DISGRACE."
```

```
·950 PRINT"[6"[DOWN]"]";TAB(8);"[
BLACK]PRESS [F1] TO PLAY AGAIN"
·952 GETK$:IFK$=""THEN952
·954 IFK$="[F1]"THENRUN2
·956 END
·1000 REM ***** DATA STARTS *****
·1002 DATA.15,60,.10,25,1.5,200,1
.5,200,2,300,.2,30,.09,15,60,450
0
·1004 DATA75,5000,300,20000,10,10
00,"BRICK","BRICKS","EACH"."CEMN
T"
·1006 DATA"CEMENT","PER LB.","WOO
D ","BOARDS","EA.","SHING"
·1008 DATA"SHINGLES","EA.","PIPES
","PIPE","EACH","WIRE ","WIRE"
·1010 DATA"PER FT.","PLSTR","PLAS
TER","PER LB.","WINDW","WINDOWS"
·1012 DATA"EACH","DOORS","DOORS",
"EACH","G.DR.","GA. DOORS","EA."
·1014 DATA"PAINT","PAINT","GALLON
",801,42,802,41,803,42,804,41,80
5
·1016 DATA42,806,41,807,42,808,41
,809,42,810,41,814,41,815,42,823
·1018 DATA42,824,41,825,42,826,41
,761,41,762,42,763,41,764,42,765
·1020 DATA41,766,42,767,41,768,42
,769,41,770,42,774,42,775,41,783
·1022 DATA41,784,42,785,41,786,42
,721,42,722,41,723,42,724,41,725
·1024 DATA42,726,41,727,42,728,41
,729,42,730,41,734,41,735,42,743
·1026 DATA42,744,41,745,42,746,41
,681,41,682,42,683,41,684,42,641
·1028 DATA42,642,41,643,42,644,41
,601,41,602,42,603,41,604,42,561
·1030 DATA42,562,41,563,42,564,41
,521,41,522,42,523,41,524,42,481
·1032 DATA42,482,41,483,42,484,41
,441,41,442,42,443,41,444,42,401
·1034 DATA33,402,41,403,42,404,23
,362,42,363,41,322,41,323,42,282
·1036 DATA42,283,41,242,41,243,42
,202,42,203,41,162,41,163,42,122
·1038 DATA42,123,41,82,41,83,42,4
2,42,43,41,2,41,3,42,-1,-1,771
·1040 DATA101,773,103,731,101,691
,101,651,101,611,101,733,103,693
·1042 DATA103,653,103,613,103,571
,79,572,119,573,80,776,101,736,1
01
·1044 DATA696,101,656,101,616,101
,782,103,742,103,702,103,662,103
·1046 DATA622,103,576,79,577,119,
578,119,579,119,580,119,581,119
·1048 DATA582,80,681,76,641,101,6
01,101,561,101,521,101,481,101,6
82
·1050 DATA111,683,111,684,111,685
,111,686,111,687,111,688,111,689
·1052 DATA111,690,111,531,101,491
,101,694,111,695,76,655,76,615,7
6
·1054 DATA575,76,535,101,495,101,
481,79,482,119,483,119,484,119,4
85
·1056 DATA119,486,119,487,119,488
,119,489,119,490,119,491,79,492
·1058 DATA119,493,119,494,119,682
,76,642,101,602,101,562,101,522
·1060 DATA101,482,79,690,76,650,7
9,610,79,570,79,530,101,490,79,6
02
·1062 DATA76,603,111,604,111,605,
111,606,111,607,111,608,111,609
·1064 DATA122,569,103,529,80,522,
79,523,119,524,119,525,119,526,1
19
·1065 DATA527,119,528,119,683,76,
643,101,684,76,644,101,685,76,64
5
·1066 DATA101,686,76,646,101,687,
76,647,101,688,76,648,101,689,76
·1068 DATA649,101,641,79,601,79,5
61,79,521,79,481,79,482,79,483,7
9
·1070 DATA484,79,485,79,486,79,48
7,79,488,79,489,79,532,101,492,7
9
·1072 DATA533,101,493,79,534,76,4
94,79,654,111,614,111,574,111,70
3
·1074 DATA122,704,122,705,122,706
,122,663,103,623,103,583,103,664
·1075 DATA103,624,103,584,103,665
,103,625,103,585,103,666,103,626
·1076 DATA103,586,103,546,80,545,
80,544,80,543,80,542,80,541,80,5
40
·1078 DATA80,539,80,538,80,537,80
,536,80,535,79,441,101,401,101,3
61
·1080 DATA101,321,101,281,101,241
,101,454,101,414,101,374,101,334
·1082 DATA101,294,101,254,101,240
```

```
,119,241,79,242,119,243,119,244
•1083 DATA119,245,119,246,119,247
,119,248,119,249,119,250,119,251
•1084 DATA119,252,119,253,119,455
,101,415,101,375,101,335,101,295
•1086 DATA101,255,101,215,101,502
,103,462,103,422,103,382,103,342
•1088 DATA103,302,103,262,103,222
,103,215,79,216,119,217,119,218
•1090 DATA119,219,119,220,119,221
,119,222,80,498,101,458,101,418
•1091 DATA101,378,101,338,101,298
,101,258,101,218,79,499,103,459
•1092 DATA103,419,103,379,103,339
,103,299,103,259,103,219,80,496
•1094 DATA101,501,103,456,101,461
,103,416,76,417,122,420,76,421,1
22
•1096 DATA376,101,377,103,380,101
,381,103,336,101,337,103,340,101
•1098 DATA341,103,296,79,297,80,3
00,79,301,80,256,101,261,103,216
•1099 DATA79,221,80,454,79,414,79
,374,79,334,79,294,79,254,79,442
•1100 DATA101,402,101,362,101,322
,101,282,101,242,79,441,79,401,7
9
•1102 DATA361,79,321,79,281,79,28
2,79,283,119,284,119,285,119,286
•1104 DATA119,287,119,288,119,289
,119,290,119,291,119,292,119,293
•1106 DATA80,333,103,373,103,413,
122,412,111,411,111,410,111,409
•1107 DATA111,408,111,407,111,406
,111,405,111,404,111,403,111,402
•1108 DATA76,445,101,405,76,365,1
01,325,101,285,79,245,79,446,101
•1110 DATA406,76,366,101,326,101,
286,79,246,79,447,101,407,76,367
•1112 DATA101,327,101,287,79,247,
79,448,101,408,76,368,101,328,10
1
•1114 DATA288,79,248,79,449,101,4
09,76,369,101,329,101,289,79,249
•1115 DATA79,450,101,410,76,370,1
01,330,101,290,79,250,79,451,101
•1116 DATA411,76,371,101,331,101,
291,79,251,79,325,76,326,76,327,
76
•1118 DATA328,76,329,76,330,76,49
5,79,496,79,497,119,498,79,499,8
0
•1120 DATA500,119,501,80,502,80,4
58,79,459,80,418,79,419,80,378,7
9
•1122 DATA379,80,338,79,339,80,29
8,79,299,80,255,79,256,79,257,11
9
•1123 DATA258,79,259,80,260,119,2
61,80,262,80,178,76,179,122,138,
79
•1124 DATA139,80,177,79,180,80,13
7,79,140,80,98,79,99,80,176,79,1
81
•1126 DATA80,59,77,100,77,141,77,
182,77,223,77,58,78,97,78,136,78
•1128 DATA175,78,214,78,45,77,86,
77,127,77,168,77,209,77,44,78,83
•1130 DATA78,122,78,161,78,200,78
,5,100,6,100,7,100,8,100,9,100,1
0
•1132 DATA100,11,100,12,100,13,10
0,14,100,15,100,16,100,17,100,18
•1134 DATA100,51,77,92,77,133,77,
174,77,50,78,89,78,128,78,167,78
•1136 DATA206,78,55,77,96,77,54,7
8,93,78,132,78,171,78,210,78,48,
77
•1138 DATA89,86,130,77,171,86,212
,77,47,78,86,86,125,78,164,78,20
3
•1140 DATA78,200,233,201,160,202,
160,203,160,204,160,205,160,206
•1142 DATA160,207,160,208,160,209
,160,210,160,211,160,212,160,213
•1144 DATA160,214,105,161,233,162
,160,163,160,164,160,165,160,166
•1146 DATA160,167,160,168,160,169
,160,170,160,171,160,172,160,173
•1148 DATA160,174,160,175,105,122
,233,123,160,124,160,125,160,126
•1150 DATA160,127,160,128,160,129
,160,130,160,131,160,132,160,133
•1151 DATA160,134,160,135,160,136
,105,83,233,84,160,85,160,86,160
•1152 DATA87,160,88,160,89,160,90
,160,91,160,92,160,93,160,94,160
•1154 DATA95,160,96,160,97,105,44
,233,5,32,45,160,6,32,46,160,7,3
2
•1156 DATA47,160,8,32,48,160,9,32
,49,160,10,32,50,160,11,32,51,16
0
•1158 DATA12,32,52,160,13,32,53,1
60,14,32,54,160,15,32,55,160,16,
32
```

```
•1160 DATA56,160,17,32,57,160,18,
 32,58,105,323,32,363,32,443,32,4
 44
•1162 DATA32,563,32,564,32,-1,-1,
 200,233,201,250,202,250,203,250
•1164 DATA204,250,205,250,206,250
 ,207,250,208,250,209,250,210,250
•1166 DATA211,250,212,250,213,250
 ,214,105,161,233,162,204,163,204
•1168 DATA164,204,165,204,166,204
 ,167,204,168,204,169,204,170,204
•1170 DATA171,204,172,204,173,204
 ,174,204,175,105,122,233,123,250
•1172 DATA124,250,125,250,126,250
 ,127,250,128,250,129,250,130,250
•1174 DATA131,250,132,250,133,250
 ,134,250,135,250,136,105,83,233,
 84
•1176 DATA204,85,204,86,204,87,20
 4,88,204,89,204,90,204,91,204,92
•1178 DATA204,93,204,94,204,95,20
 4,96,204,97,105,44,233,45,250,46
•1180 DATA250,47,250,48,250,49,25
 0,50,250,51,250,52,250,53,250,54
•1182 DATA250,55,250,56,250,57,25
 0,58,105,223,77,182,77,141,77,10
 0
•1184 DATA77,59,77,-1,-1,681,682,
 683,684,685,686,687,688,689,690,
 641
•1186 DATA650,601,561,521,610,570
 ,530,531,532,533,534,694,695,654
•1188 DATA655,614,615,574,575,704
 ,705,706,664,665,666,624,625,626
•1190 DATA584,585,586,544,545,546
 ,481,482,483,484,485,486,487,488
•1192 DATA489,490,491,492,493,441
 ,442,443,444,445,446,447,448,449
•1194 DATA450,451,452,453,405,406
 ,407,408,409,410,365,366,367,368
•1196 DATA369,370,325,326,327,328
 ,329,330,285,286,287,288,289,290
•1198 DATA401,361,321,281,241,242
 ,243,244,245,246,247,248,249,250
•1200 DATA251,252,253,495,496,497
 ,498,499,500,501,502,455,456,457
•1202 DATA458,459,460,461,462,255
 ,256,257,258,259,260,261,262,215
•1203 DATA216,217,218,219,220,221
 ,222,415,375,335,295,422,382,342
•1204 DATA302,418,419,378,379,338
 ,339,298,299,176,180,181,140,98,
 99
•1206 DATA-1,535,247,536,247,537,
 247,538,247,539,247,540,247,541
•1208 DATA247,542,247,543,247,544
 ,247,545,247,546,247,703,231,663
•1210 DATA231,623,231,583,231,543
 ,208,494,231,454,231,414,231,374
•1212 DATA231,334,231,294,231,254
 ,231,602,32,603,32,604,32,605,32
•1214 DATA606,32,607,32,608,32,60
 9,32,642,247,643,247,644,247,645
•1215 DATA247,646,247,647,247,648
 ,247,649,247,562,32,569,32,522,3
 2
•1216 DATA523,32,524,32,525,32,52
 6,32,527,32,528,32,529,32,138,32
•1218 DATA139,32,178,32,179,32,13
 7,231,177,231,240,32,-1,-1,602,2
 07
•1220 DATA603,208,562,207,563,208
 ,522,207,523,208,608,207,609,208
•1222 DATA568,207,569,208,528,207
 ,529,208,604,160,605,160,606,160
•1224 DATA607,160,564,207,569,208
 ,528,207,529,208,604,160,605,160
•1226 DATA606,160,607,160,564,160
 ,565,160,566,160,567,160,524,247
•1228 DATA525,247,526,247,527,247
 ,402,204,403,239,404,250,362,207
•1230 DATA363,247,364,208,322,229
 ,323,160,324,231,282,207,283,247
•1232 DATA284,208,411,204,412,239
 ,413,250,371,207,372,247,373,208
•1233 DATA331,229,332,160,333,231
 ,291,207,292,247,293,208,416,204
•1234 DATA417,250,376,207,377,208
 ,336,229,337,231,296,207,297,208
•1236 DATA420,204,421,250,380,207
 ,381,208,340,229,341,231,300,207
•1238 DATA301,208,178,250,179,250
 ,138,208,139,208,-1,-1,495,496,4
 97
•1242 DATA498,499,500,501,502,503
 ,504,505,506,801,802,803,804,805
•1244 DATA806,807,808,809,810,814
 ,815,823,824,825,826,-1,571,207
•1246 DATA611,229,651,229,691,229
 ,731,229,771,229,572,247,612,111
•1248 DATA652,32,692,160,732,160,
 772,160,573,208,613,231,653,231
•1250 DATA693,215,733,231,773,231
 ,-1,-1,576,207,577,247,578,247,5
 79
•1252 DATA247,580,247,581,247,582
```

IMPORTANT

Before typing in an *Ahoy!* program, refer to the first two pages of the program listings section.

```
,208,616,207,617,247,618,247,619
•1254 DATA247,620,247,621,247,622
,208,656,207,657,247,658,247,659
•1256 DATA247,660,247,661,247,662
,208,696,207,697,247,698,247,699
•1258 DATA247,700,247,701,247,702
,208,736,207,737,247,738,247,739
•1260 DATA247,740,247,741,247,742
,208,776,207,777,247,778,247,779
•1262 DATA247,780,247,781,247,782
,208,-1,-1,1,2,3,6,7,8,9,10,11,1
2
•1264 DATA-1,641,642,643,644,645,
646,647,648,649,650,690,610,570,
530
•1266 DATA601,561,521,681,682,683
,684,685,686,687,688,689,531,532
•1268 DATA533,534,574,575,614,615
,654,655,694,695,535,536,537,538
•1270 DATA539,540,541,542,543,583
,623,663,703,704,705,706,664,665
•1272 DATA665,666,624,625,626,584
,585,586,544,545,546,481,482,483
•1274 DATA484,485,486,487,488,489
,490,491,492,493,494,441,442,443
•1276 DATA444,445,401,361,321,281
,241,242,243,244,245,285,325,365
•1278 DATA405,450,451,452,453,454
,410,370,330,290,250,251,252,253
•1280 DATA254,294,334,374,414,446
,447,448,449,406,407,408,409,366
•1282 DATA367,368,369,326,327,328
,329,286,287,288,289,246,247,248
•1284 DATA249,455,456,457,458,459
,460,461,462,415,375,335,295,422
•1286 DATA382,342,302,255,215,176
,137,177,98,99,140,180,181,222,2
62
•1288 DATA256,257,258,259,260,261
,216,217,218,219,220,221,418,419
•1290 DATA378,379,338,339,298,299
,-1,571,611,651,691,731,771,572
•1292 DATA612,652,692,732,772,573
,613,653,693,733,773,576,577,578
•1294 DATA579,580,581,582,616,617
,618,619,620,621,622,656,657,658
•1296 DATA659,660,661,662,696,697
,698,699,700,701,702,736,737,738
•1298 DATA739,740,741,742,776,777
,778,779,780,781,782,-1
```

## BUG REPELLENT LINE CODES FOR CONSTRUCTION CO.

```
LINE #   2:AA        LINE #  258:EP
LINE #   4:KP        LINE #  260:BE
LINE #   6:CB        LINE #  300:AF
LINE #   8:KP        LINE #  302:GM
LINE #  10:OJ        LINE #  304:OE
LINE #  12:KP        LINE #  306:PE
LINE #  14:IG        LINE #  350:EF
LINE #  16:KP        LINE #  352:CO
LINE #  18:IH        LINE #  400:GN
LINE #  20:HM        LINE #  402:GN
LINE #  22:DL        LINE #  500:HN
LINE #  24:NO        LINE #  510:DD
LINE #  26:DP        LINE #  600:NL
LINE #  28:MN        LINE #  602:NI
LINE #  30:GL        LINE #  604:LI
LINE #  32:FP        LINE #  606:FO
LINE #  34:LB        LINE #  608:KJ
LINE #  36:GK        LINE #  610:IF
LINE #  38:LM        LINE #  612:MD
LINE #  40:LN        LINE #  614:MM
LINE # 100:NI        LINE #  616:HE
LINE # 102:EC        LINE #  618:KJ
LINE # 104:DK        LINE #  620:GM
LINE # 150:AJ        LINE #  622:JF
LINE # 152:CH        LINE #  624:GP
LINE # 154:OB        LINE #  626:PN
LINE # 156:CA        LINE #  628:CH
LINE # 158:KD        LINE #  630:FC
LINE # 160:GK        LINE #  632:GK
LINE # 164:NM        LINE #  634:KI
LINE # 190:PC        LINE #  636:LN
LINE # 200:LE        LINE #  700:HB
LINE # 202:GA        LINE #  702:JP
LINE # 204:IC        LINE #  704:JD
LINE # 206:AG        LINE #  800:JC
LINE # 208:FN        LINE #  802:LB
LINE # 210:NF        LINE #  804:LJ
LINE # 212:AL        LINE #  806:HK
LINE # 214:JF        LINE #  808:NM
LINE # 216:AE        LINE #  810:BD
LINE # 218:GD        LINE #  812:CH
LINE # 220:GF        LINE #  814:LF
LINE # 222:FC        LINE #  816:FI
LINE # 224:OP        LINE #  818:DI
LINE # 226:IK        LINE #  820:MJ
LINE # 228:OL        LINE #  822:GF
LINE # 250:CA        LINE #  824:CO
LINE # 252:HL        LINE #  826:NA
LINE # 254:KA        LINE #  828:GJ
LINE # 256:LJ        LINE #  830:AG
```

```
LINE # 832:FD
LINE # 834:IP
LINE # 836:CH
LINE # 838:DG
LINE # 840:AF
LINE # 842:JL
LINE # 844:DE
LINE # 846:LF
LINE # 848:BG
LINE # 850:PD
LINE # 852:DG
LINE # 854:LF
LINE # 856:BG
LINE # 858:DE
LINE # 860:MN
LINE # 862:GF
LINE # 864:OF
LINE # 866:NN
LINE # 868:OE
LINE # 870:PI
LINE # 872:JO
LINE # 874:CH
LINE # 876:IP
LINE # 878:LJ
LINE # 880:GN
LINE # 882:GK
LINE # 884:FB
LINE # 886:PD
LINE # 888:HB
LINE # 890:JA
LINE # 892:EE
LINE # 894:MG
LINE # 896:NM
LINE # 898:HG
LINE # 900:KI
LINE # 902:JK
LINE # 904:HB
LINE # 906:IF
LINE # 908:PM
LINE # 910:AM
LINE # 912:AP
LINE # 914:JC
LINE # 916:AF
LINE # 918:GO
LINE # 920:LB
LINE # 922:NG
LINE # 924:JO
LINE # 926:AE
LINE # 928:DO
LINE # 930:IP
LINE # 932:LI
LINE # 934:DL
LINE # 936:MO
LINE # 938:GI
LINE # 940:KA
LINE # 942:OO
LINE # 944:KI
LINE # 946:KA
LINE # 948:KA
LINE # 950:GE
LINE # 952:JD
LINE # 954:MI
LINE # 956:IC
LINE # 1000:HA
LINE # 1002:HH
LINE # 1004:JO
LINE # 1006:ON
LINE # 1008:PD
LINE # 1010:OM
LINE # 1012:EP
LINE # 1014:FN
LINE # 1016:NL
LINE # 1018:NP
LINE # 1020:PD
LINE # 1022:BO
LINE # 1024:AD
LINE # 1026:CD
LINE # 1028:IL
LINE # 1030:EH
LINE # 1032:PC
LINE # 1034:DH
LINE # 1036:GP
LINE # 1038:BD
LINE # 1040:JH
LINE # 1042:IA
LINE # 1044:BK
LINE # 1046:HG
LINE # 1048:CD
LINE # 1050:EN
LINE # 1052:PC
LINE # 1054:MM
LINE # 1056:JK
LINE # 1058:BN
LINE # 1060:IO
LINE # 1062:KP
LINE # 1064:LB
LINE # 1065:ND
LINE # 1066:FJ
LINE # 1068:OO
LINE # 1070:OF
LINE # 1072:ML
LINE # 1074:DG
LINE # 1075:GH
LINE # 1076:AK
LINE # 1078:KI
LINE # 1080:DC
LINE # 1082:DL
LINE # 1083:GB
LINE # 1084:CO
LINE # 1086:FD
LINE # 1088:OL
LINE # 1090:PA
LINE # 1091:CL
LINE # 1092:KP
LINE # 1094:KB
LINE # 1096:MO
LINE # 1098:ID
LINE # 1099:MJ
LINE # 1100:KD
LINE # 1102:JC
LINE # 1104:BB
LINE # 1106:MP
LINE # 1107:AK
LINE # 1108:JM
LINE # 1110:OO
LINE # 1112:OA
LINE # 1114:CF
LINE # 1115:HD
LINE # 1116:MK
LINE # 1118:FE
LINE # 1120:PE
LINE # 1122:II
LINE # 1123:HL
LINE # 1124:CH
LINE # 1126:NL
LINE # 1128:IP
LINE # 1130:NP
LINE # 1132:JO
LINE # 1134:DA
LINE # 1136:HJ
LINE # 1138:OC
LINE # 1140:CH
LINE # 1142:GP
LINE # 1144:FP
LINE # 1146:IG
LINE # 1148:HB
LINE # 1150:KC
LINE # 1151:IH
LINE # 1152:HJ
LINE # 1154:OI
LINE # 1156:GP
LINE # 1158:LN
LINE # 1160:JO
LINE # 1162:IJ
LINE # 1164:OH
LINE # 1166:MO
LINE # 1168:KL
LINE # 1170:NJ
LINE # 1172:KC
LINE # 1174:NK
LINE # 1176:MD
LINE # 1178:IM
LINE # 1180:OJ
LINE # 1182:OO
LINE # 1184:BD
LINE # 1186:HD
LINE # 1188:KA
LINE # 1190:OO
LINE # 1192:NG
LINE # 1194:HM
LINE # 1196:LK
LINE # 1198:JK
LINE # 1200:JA
LINE # 1202:DB
LINE # 1203:PL
LINE # 1204:KN
LINE # 1206:IO
LINE # 1208:LO
LINE # 1210:HG
LINE # 1212:NG
LINE # 1214:AJ
LINE # 1215:DB
LINE # 1216:EH
LINE # 1218:CC
LINE # 1220:FH
LINE # 1222:MG
LINE # 1224:PJ
LINE # 1226:NP
LINE # 1228:AM
LINE # 1230:NL
LINE # 1232:GE
LINE # 1233:LM
LINE # 1234:BL
LINE # 1236:NC
LINE # 1238:BO
LINE # 1242:HO
LINE # 1244:MK
LINE # 1246:DA
LINE # 1248:GN
LINE # 1250:DI
LINE # 1252:LI
LINE # 1254:JF
LINE # 1256:LJ
LINE # 1258:NK
LINE # 1260:OO
LINE # 1262:FM
LINE # 1264:BE
LINE # 1266:JJ
LINE # 1268:AP
LINE # 1270:HM
LINE # 1272:KO
LINE # 1274:NA
```

IMPORTANT

Before typing in an *Ahoy!* program, refer to the first two pages of the program listings section.

```
LINE # 1276:AN
LINE # 1278:OK
LINE # 1280:DE
LINE # 1282:HO
LINE # 1284:PI
LINE # 1286:GN
LINE # 1288:KB
LINE # 1290:NM
LINE # 1292:EL
LINE # 1294:NI
LINE # 1296:KG
LINE # 1298:DJ
LINES: 324
```

# CROSS REF

FROM PAGE 98

```
•1 REM:::RUPERT REPORT #12:::
•2 GOTO 10000
•3 :
•4 :REM....SET MAX LINE# IN 10250
•5 :
•6 :REM..LINES 7-12 ARE DEMOS ONL
Y
•7 IF N THEN 4 : GOSUB 5
•8 IF N THEN GOTO 5
•9 IF N THEN GOSUB 6
•10 ON N GOTO 7,8,9,10
•11 ON N GOSUB 9,10,4
•12 GOTO 5:GOSUB 10:ON N GOTO 3,4
•10000 :REM======================
•10001 :REM   CROSS-REFERENCE
•10002 :REM     GOSUBS AND GOTOS
•10003 :REM======================
•10004 :
•10010 GOSUB 10200 :REM INITIALIZ
E
•10020 GOSUB 10400 :REM SCAN A LI
NE
•10030 GOSUB 10600 :REM UPDATE PT
RS
•10040 IF DNFLG=0 THEN 10020
•10050 GOSUB 20000 :REM PRINT RES
ULTS
•10060 GOSUB 20700 :REM BACK REFE
RENCE
•10070 CLOSE FF
•10099 END
•10199 :REM----------------------
-
•10200 :REM......INITIALIZE......
.
•10201 :REM----------------------
-
•10205 PRINT CHR$(147) :REM CLEAR
•10210 DEF FNPT(X)=PEEK(X)+256*PE
EK(X+1)
•10220 DEF FNMS(X)=INT(X/256)
•10230 DEF FNLS(X)=X-256*FNMS(X)
•10240 DIM LN(200,11)
•10245 :
•10250 MX=9999 :REM MAXIMUM LINE#
•10255 :
•10260 :REM 'GOTO,GOSUB,THEN' COD
ES
•10270 GT=137:GS=141:TH=167
•10280 :REM  CL=CURRENT LINE ADDR
•10290 :REM  NN=PRGM LINE COUNT
•10300 CL=FNPT(43) :REM FIRST LIN
E
•10310 NN=1
•10320 DNFLG=0
•10325 PRINT CHR$(147) :REM CLEAR
•10330 INPUT"SEND OUTPUT TO SCREE
N OR PRINTER (S/P)";RR$
•10340 IF RR$="S" THEN DEV=3:GOTO
 10370
•10350 IF RR$="P" THEN DEV=4:GOTO
 10370
•10360 GOTO 10330
•10365 :REM SPECIFY FILE# FOR FF
•10370 FF=222: OPEN FF,DEV
•10380 PRINT CHR$(147)"SCANNING L
INE # :"
•10390 RETURN
•10399 :REM----------------------
-
•10400 :REM...SCAN NEXT LINE.....
•10401 :REM----------------------
-
•10405 :REM  NL=NEXT LINE ADDR
•10410 NL=FNPT(CL)
•10420 A1=CL+4 :REM START OF TEXT
•10430 A2=NL-2 :REM END OF TEXT
•10440 LN(NN,0)=FNPT(CL+2):REM LI
NE#
•10445 :REM PUT CRSR @ ROW 0,COL
 20
•10450 POKE214,255:PRINT:POKE211,
 20
•10460 PRINT LN(NN,0);
•10470 LN(NN,1)=0 :REM COUNT
•10475 :REM STEP THRU ENTIRE LINE
•10480 FOR MM=A1 TO A2-2
•10490 PM=PEEK(MM)
•10495 :REM  SKIP LINE AFTER 'REM
'
•10500 IF PM=143 THEN MM=A2-2 :GO
TO 10530
```

```
•10510 IF PM=TH THEN GOSUB 20200
•10520 IF PM=GT OR PM=GS THEN GOS
 UB 20400
•10530 NEXT MM
•10540 RETURN
•10599 :REM----------------------
 -
•10600 :REM...UPDATE POINTERS...
•10601 :REM----------------------
 -
•10610 CL=NL
•10620 IF FNPT(CL)=0 THEN DNFLG=1
    :GOTO 10660
•10630 IF FNPT(CL+2)>MX THEN DNFL
 G=1 :GOTO 10660
•10640 NN=NN+1
•10650 CT=0
•10660 RETURN
•10999 :REM----------------------
 -
•20000 :REM...PRINT RESULTS...
•20001 :REM----------------------
 -
•20005 PRINT#FF:PRINT#FF,"[11"="]
 "
•20010 FOR XX=1 TO NN
•20020 CT=LN(XX,1)
•20030 IF CT=0 THEN 20090
•20040 PRINT#FF,LN(XX,0)"==>",
•20050 FOR YY=1 TO CT
•20060 PRINT#FF, LN(XX,YY+1);
•20070 NEXT YY
•20080 PRINT#FF
•20090 NEXT XX
•20100 RETURN
•20199 :
•20200 :REM...FOUND 'THEN'.....
•20201 :
•20210 MM=MM+1 : PM=PEEK(MM)
•20220 IF PM=32 THEN 20210:REM SP
 ACE
•20230 IF PM<48 OR PM>57 THEN 202
 80
•20240 GOSUB 20600 :REM LL <== LI
 NE#
•20250 CT=CT+1
•20260 LN(NN,1)=CT
•20270 LN(NN,CT+1)=LL
•20280 RETURN
•20399 :
•20400 :REM..HAVE 'GOSUB' OR 'GOT
 O'
•20401 :
•20410 MM=MM+1 : PM=PEEK(MM)
•20420 IF PM=32 THEN 20410
•20430 GOSUB 20600 :REM LL <==LIN
 E#
•20440 CT=CT+1
•20450 LN(NN,1)=CT
•20460 LN(NN,CT+1)=LL
•20465 :REM  ON GOSUB/GOTO N1,N2,
 .
•20470 IF PM=ASC(",") THEN 20410
•20480 RETURN
•20599 :
•20600 :REM.PUT LINE# @ MM INTO L
 L
•20601 :
•20610 LL$=""
•20620 LL$=LL$+CHR$(PM)
•20630 MM=MM+1 : PM=PEEK(MM)
•20640 IF PM>47 AND PM<58 THEN 20
 620
•20650 IF PM=32 THEN MM=MM+1:PM=P
 EEK(MM):GOTO 20650
•20660 LL=VAL(LL$)
•20670 RETURN
•20699 :REM----------------------
 -
•20700 :REM...PRINT BACK REFERENC
 ES
•20701 :REM----------------------
 -
•20705 PRINT#FF:PRINT#FF,"[11"="]
 "
•20710 FOR XX=1 TO NN
•20720 SS=LN(XX,0) :REM SEARCH LI
 NE
•20740 FOR YY=1 TO NN
•20750 CT=LN(YY,1):IF CT=0 THEN 2
 0820
•20760 FOR ZZ=2 TO CT+1
•20770 IF LN(YY,ZZ)<>SS THEN 2081
 0
•20780 IF TT=0 THEN PRINT#FF,SS "
 <==",
•20790 TT=1
•20800 PRINT#FF, LN(YY,0);
•20810 NEXT ZZ
•20820 NEXT YY
•20830 IF TT>0 THEN PRINT#FF : TT
 =0
•20840 NEXT XX
•20850 RETURN
```

IMPORTANT

Before typing in an *Ahoy!* program, refer to the first two pages of the program listings section.


## C-64 BUG REPELLENT LINE CODES FOR CROSS REF

```
LINE # 1:JJ
LINE # 2:IE
LINE # 3:DI
LINE # 4:FK
LINE # 5:DI
LINE # 6:IJ
LINE # 7:MN
LINE # 8:EC
LINE # 9:EN
LINE # 10:AA
LINE # 11:JM
LINE # 12:DN
LINE # 10000:MO
LINE # 10001:BP
LINE # 10002:BO
LINE # 10003:MO
LINE # 10004:DI
LINE # 10010:FL
LINE # 10020:AB
LINE # 10030:ED
LINE # 10040:JK
LINE # 10050:HC
LINE # 10060:DH
LINE # 10070:CL
LINE # 10099:IC
LINE # 10199:OG
LINE # 10200:FN
LINE # 10201:OG
LINE # 10205:IN
LINE # 10210:OL
LINE # 10220:ND
LINE # 10230:GO
LINE # 10240:II
LINE # 10245:DI
LINE # 10250:ND
LINE # 10255:DI
LINE # 10260:NO
LINE # 10270:AF
LINE # 10280:AG
LINE # 10290:HJ
LINE # 10300:AN
LINE # 10310:IC
LINE # 10320:FM
LINE # 10325:IN
LINE # 10330:IM
LINE # 10340:MN
LINE # 10350:MD
LINE # 10360:IC
LINE # 10365:IJ
LINE # 10370:EA
LINE # 10380:AK
LINE # 10390:IM
LINE # 10399:OG
LINE # 10400:PL
LINE # 10401:OG
LINE # 10405:FA
LINE # 10410:GP
LINE # 10420:JL
LINE # 10430:BG
LINE # 10440:JE
LINE # 10445:LJ
LINE # 10450:LC
LINE # 10460:MN
LINE # 10470:CA
LINE # 10475:GC
LINE # 10480:GG
LINE # 10490:OL
LINE # 10495:PH
LINE # 10500:HD
LINE # 10510:KA
LINE # 10520:GJ
LINE # 10530:CH
LINE # 10540:IM
LINE # 10599:OG
LINE # 10600:IK
LINE # 10601:OG
LINE # 10610:NL
LINE # 10620:AP
LINE # 10630:CA
LINE # 10640:NG
LINE # 10650:HI
LINE # 10660:IM
LINE # 10999:OG
LINE # 20000:FF
LINE # 20001:OG
LINE # 20005:LN
LINE # 20010:FF
LINE # 20020:DI
LINE # 20030:LL
LINE # 20040:LP
LINE # 20050:GC
LINE # 20060:NA
LINE # 20070:DP
LINE # 20080:CD
LINE # 20090:DF
LINE # 20100:IM
LINE # 20199:DI
LINE # 20200:BI
LINE # 20201:DI
LINE # 20210:PI
LINE # 20220:CF
LINE # 20230:HK
LINE # 20240:NL
LINE # 20250:LI
LINE # 20260:DI
LINE # 20270:LF
LINE # 20280:IM
LINE # 20399:DI
LINE # 20400:II
LINE # 20401:DI
LINE # 20410:PI
LINE # 20420:NC
LINE # 20430:NL
LINE # 20440:LI
LINE # 20450:DI
LINE # 20460:LF
LINE # 20465:EB
LINE # 20470:PG
LINE # 20480:IM
LINE # 20599:DI
LINE # 20600:JC
LINE # 20601:DI
LINE # 20610:LC
LINE # 20620:GJ
LINE # 20630:PI
LINE # 20640:DP
LINE # 20650:GB
LINE # 20660:CO
LINE # 20670:IM
LINE # 20699:OG
LINE # 20700:LA
LINE # 20701:OG
LINE # 20705:LN
LINE # 20710:FF
LINE # 20720:FF
LINE # 20740:GD
LINE # 20750:EF
LINE # 20760:CH
LINE # 20770:MJ
LINE # 20780:NO
LINE # 20790:IO
LINE # 20800:GO
LINE # 20810:DN
LINE # 20820:DP
LINE # 20830:JJ
LINE # 20840:DF
LINE # 20850:IM
LINES: 147
```

# CHARACTER SET EXAMPLE

**FROM PAGE 98** C-64 VERSION

```
•6 REM SAMPLE CHARACTER SET HANDL
ER
•7 REM
•8 REM HOW MANY CHARACTERS?
•9 REM
•10 Z=32
•12 REM
•13 REM ARRAY TO HOLD COMPLEX CHA
RACTERS
•14 REM
•15 DIM CC$(7)
•17 REM
•18 REM GO EXECUTE SUBROUTINE
•19 REM
•20 GOSUB 400
•27 REM
•28 REM CURSOR CONTROL CHARACTERS
•29 REM
•30 D$=CHR$(17):L$=CHR$(157):U$=C
HR$(145):R$=CHR$(29)
•37 REM
•38 REM F1=BIG SHIP  F3=SMALL SHI
P  F5=STARSHIP 1
```

```
39 REM
40 CC$(0)="KLM"+D$+L$+L$+L$+"NOP
 "+U$:CC$(1)=R$+"J"+D$+L$+L$+"HI"
 +U$:CC$(2)="STU"
47 REM
48 REM F7=STARSHIP 2  F2=STARSHI
 P 3  F6=COACH
49 REM
50 CC$(3)="VWX":CC$(4)="12"+D$+L
 $+L$+"YZ"+U$:CC$(6)=R$+"6"+D$+L$
 +L$+"345"+U$
57 REM
58 REM F4=STARBASE
59 REM
60 CC$(5)=R$+R$+CC$(4)+D$+D$++L$
 +L$+L$+L$+CC$(2)+CC$(3)+U$+U$
67 REM
68 REM 4-WAY INTERSECTION
69 REM
70 Q$=D$+L$+L$+L$+L$:A$=R$+"AA"+
 R$+Q$:CC$(7)=R$+"FG"+R$+Q$+"FAAG
 "+Q$+"ACCA"
75 CC$(7)=CC$(7)+Q$+"EAAD"+Q$+A$
 +A$+A$+A$+A$+"FBBG"+U$+U$+U$+U$+
 U$+U$+U$+U$+U$
80 PRINT "[CLEAR]":RESTORE:FOR I
 =0 TO Z:READ A$:READ A,A,A,A,A,A
 ,A,A
85 PRINT A$"=[RVSON]"A$"[RVSOFF]
 ";:NEXT
87 REM
88 REM GET VALID KEYPRESS
89 REM
90 GET A$:IF A$="" THEN 90
97 REM
98 REM INVERSE ON OR OFF
99 REM
100 A=ASC(A$):IF A=18 OR A=146 T
 HEN I$=A$:GOTO 90
107 REM
108 REM FUNCTION KEYS--PRINT COM
 PLEX CHARACTERS
109 REM
110 IF A>132 AND A<141 THEN A=A-
 133:PRINT I$CC$(A);:GOTO 90
117 REM
118 REM DON'T PRINT INVERSE SPAC
 E!
119 REM
120 IF A=32 THEN PRINT "[RVSOFF]
 "A$;:GOTO 90
127 REM
128 REM PRINT ANY OTHER KEY
129 REM
130 PRINT I$A$;:GOTO 90
397 REM
398 REM SWITCH BANKS AND COPY NO
 N-INVERSE CHARACTER PATTERNS
399 REM
400 CM=14:CS=CM*1024
410 POKE 56333,127
420 POKE 1,PEEK(1) AND 251
430 FOR I=0 TO 1023:POKE CS+I,PE
 EK(53248+I):NEXT
440 POKE 1,PEEK(1) OR 4
450 POKE 56333,129
457 REM
458 REM FLIP TO NEW CHARACTER SE
 T AREA
459 REM
460 POKE 53272,(PEEK(53272)AND 2
 40) OR CM
467 REM
468 REM ADDING 1024 PUTS IT IN I
 NVERSE VIDEO AREA OF MEMORY
469 REM
470 CS=CS+1024:FOR I=0 TO Z:READ
 C$
477 REM
478 REM CONVERT CHARACTER TO SCR
 EEN CODE
479 REM
480 C=ASC(C$):IF (C>63 AND C<96)
  OR (C>159 AND C<192) THEN C=C-6
 4:GOTO 500
490 IF C>95 AND C<128 THEN C=C-3
 2:GOTO 500
500 CP=CS+8*C
507 REM
508 REM READ PATTERN AND POKE IT
  INTO MEMORY
509 REM
510 FOR K=CP TO CP+7:READ A:POKE
  K,A:NEXT:NEXT
520 RETURN
62997 REM
62998 REM CHARACTER DATA
62999 REM
63000 DATA "A",0,251,251,251,0,1
 91,191,191
63001 DATA "B",255,255,255,255,2
 55,255,255,255
63002 DATA "C",0,0,0,0,0,191,191
 ,191
63003 DATA "D",239,238,236,232,1
 6,224,192,128
```

**IMPORTANT**
Before typing in an *Ahoy!* program, refer to the first two pages of the program listings section.

```
•63004 DATA "E",247,119,55,23,8,7
 ,3,1
•63005 DATA "F",1,2,6,14,16,47,11
 1,239
•63006 DATA "G",128,192,224,240,8
 ,236,238,239
•63007 DATA "H",0,0,0,0,120,56,31
 ,31
•63008 DATA "I",240,176,240,160,2
 07,142,252,248
•63009 DATA "J",0,0,0,0,128,128,1
 92,160
•63010 DATA "K",0,1,1,1,3,5,15,13
•63011 DATA "L",8,8,24,40,120,40,
 24,8
•63012 DATA "M",0,128,244,172,216
 ,184,248,184
•63013 DATA "N",15,5,3,1,1,255,63
 ,7
•63014 DATA "O",24,40,120,40,24,8
 ,255,255
•63015 DATA "P",248,188,254,128,1
 28,255,252,252
•63016 DATA "Q",16,24,28,22,31,16
 ,255,63
•63017 DATA "R",0,0,8,62,62,54,8,
 127
•63018 DATA "S",0,0,0,0,7,31,255,
 0
•63019 DATA "T",0,0,3,0,192,131,2
 55,0
•63020 DATA "U",0,255,252,96,192,
 128,255,0
•63021 DATA "V",0,255,63,6,3,1,25
 5,0
•63022 DATA "W",0,0,192,0,3,193,2
 55,0
•63023 DATA "X",0,0,0,0,224,248,2
 55,0
•63024 DATA "Y",2,34,98,99,113,12
 0,111,96
•63025 DATA "Z",64,68,70,198,142,
 30,246,6
•63026 DATA "1",1,1,1,3,2,2,2,2
•63027 DATA "2",128,128,128,192,6
 4,64,64,64
•63028 DATA "3",0,6,1,28,34,34,34
 ,28
•63029 DATA "4",255,126,60,189,18
 9,129,255,0
•63030 DATA "5",0,96,128,56,68,68
 ,68,56
•63031 DATA "6",0,195,189,126,102
 ,102,102,102
•63040 DATA " ",255,255,195,195,1
 95,195,255,255
•63900 OPEN2,8,2,"64,S,W":CMD2:LI
 ST-63900
```

## VIC 20 VERSION

```
•10 Z=32
•15 DIM CC$(7)
•20 GOSUB 400
•30 D$=CHR$(17):L$=CHR$(157):U$=C
 HR$(145):R$=CHR$(29)
•40 CC$(0)="KLM"+D$+L$+L$+L$+"NOP
 "+U$:CC$(1)=R$+"J"+D$+L$+L$+"HI"
 +U$:CC$(2)="STU"
•50 CC$(3)="VWX":CC$(4)="12"+D$+L
 $+L$+"YZ"+U$:CC$(6)=R$+"6"+D$+L$
 +L$+"345"+U$
•60 CC$(5)=R$+R$+CC$(4)+D$+D$++L$
 +L$+L$+L$+CC$(2)+CC$(3)+U$+U$
•70 CC$(7)="J"
•80 PRINT "[CLEAR]"
•90 GET A$:IF A$="" THEN 90
•100 A=ASC(A$):IF A=18 OR A=146 T
 HEN I$=A$:GOTO 90
•110 IF A>132 AND A<141 THEN A=A-
 133:PRINT I$CC$(A);:GOTO 90
•120 IF A=32 THEN PRINT "[RVSOFF]
 "A$;:GOTO 90
•130 PRINT I$A$;:GOTO 90
•400 CS=6144:CM=14
•460 POKE 36869,(PEEK(36869)AND 2
 40) OR CM
•470 FOR I=0 TO Z:READ C$
•480 C=ASC(C$):IF (C>63 AND C<96)
  OR (C>159 AND C<192) THEN C=C-6
 4:GOTO 500
•490 IF C>95 AND C<128 THEN C=C-3
 2:GOTO 500
•500 CP=CS+8*C
•510 FOR K=CP TO CP+7:READ A:POKE
  K,A:NEXT:NEXT
•520 RETURN
•63000 DATA "A",0,251,251,251,0,1
 91,191,191
•63001 DATA "B",255,255,255,255,2
 55,255,255,255
•63002 DATA "C",0,0,0,0,0,191,191
 ,191
•63003 DATA "D",239,238,236,232,1
 6,224,192,128
•63004 DATA "E",247,119,55,23,8,7
```

```
,3,1
•63005 DATA "F",1,2,6,14,16,47,11
1,239
•63006 DATA "G",128,192,224,240,8
,236,238,239
•63007 DATA "H",0,0,0,0,120,56,31
,31
•63008 DATA "I",240,176,240,160,2
07,142,252,248
•63009 DATA "J",0,0,0,0,128,128,1
92,160
•63010 DATA "K",0,1,1,1,3,5,15,13
•63011 DATA "L",8,8,24,40,120,40,
24,8
•63012 DATA "M",0,128,244,172,216
,184,248,184
•63013 DATA "N",15,5,3,1,1,255,63
,7
•63014 DATA "O",24,40,120,40,24,8
,255,255
•63015 DATA "P",248,188,254,128,1
28,255,252,252
•63016 DATA "Q",16,24,28,22,31,16
,255,63
•63017 DATA "R",0,0,8,62,62,54,8,
127
•63018 DATA "S",0,0,0,0,7,31,255,
0
•63019 DATA "T",0,0,3,0,192,131,2
55,0
•63020 DATA "U",0,255,252,96,192,
128,255,0
•63021 DATA "V",0,255,63,6,3,1,25
5,0
•63022 DATA "W",0,0,192,0,3,193,2
55,0
•63023 DATA "X",0,0,0,0,224,248,2
55,0
•63024 DATA "Y",2,34,98,99,113,12
0,111,96
•63025 DATA "Z",64,68,70,198,142,
30,246,6
•63026 DATA "1",1,1,1,3,2,2,2,2
•63027 DATA "2",128,128,128,192,6
4,64,64,64
•63028 DATA "3",0,6,1,28,34,34,34
,28
•63029 DATA "4",255,126,60,189,18
9,129,255,0
•63030 DATA "5",0,96,128,56,68,68
,68,56
•63031 DATA "6",0,195,189,126,102
,102,102,102
•63040 DATA " ",0,0,0,0,0,0,0,0
```

## BUG REPELLENT LINE CODES FOR C-64 CHARACTER SET EXAMPLE

```
LINE # 6:ME        LINE # 118:FI
LINE # 7:JD        LINE # 119:JD
LINE # 8:KI        LINE # 120:LD
LINE # 9:JD        LINE # 127:JD
LINE # 10:HC       LINE # 128:GJ
LINE # 12:JD       LINE # 129:JD
LINE # 13:CG       LINE # 130:OC
LINE # 14:JD       LINE # 397:JD
LINE # 15:LD       LINE # 398:KP
LINE # 17:JD       LINE # 399:JD
LINE # 18:GF       LINE # 400:FF
LINE # 19:JD       LINE # 410:HL
LINE # 20:CE       LINE # 420:IM
LINE # 27:JD       LINE # 430:IF
LINE # 28:FH       LINE # 440:BE
LINE # 29:JD       LINE # 450:HN
LINE # 30:FP       LINE # 457:JD
LINE # 37:JD       LINE # 458:DD
LINE # 38:FI       LINE # 459:JD
LINE # 39:JD       LINE # 460:NJ
LINE # 40:MI       LINE # 467:JD
LINE # 47:JD       LINE # 468:FM
LINE # 48:AO       LINE # 469:JD
LINE # 49:JD       LINE # 470:GF
LINE # 50:AB       LINE # 477:JD
LINE # 57:JD       LINE # 478:NF
LINE # 58:JM       LINE # 479:JD
LINE # 59:JD       LINE # 480:OF
LINE # 60:JM       LINE # 490:JN
LINE # 67:JD       LINE # 500:MC
LINE # 68:FK       LINE # 507:JD
LINE # 69:JD       LINE # 508:EA
LINE # 70:FO       LINE # 509:JD
LINE # 75:NB       LINE # 510:FK
LINE # 80:EL       LINE # 520:IM
LINE # 85:OG       LINE # 62997:JD
LINE # 87:JD       LINE # 62998:EG
LINE # 88:EN       LINE # 62999:JD
LINE # 89:JD       LINE # 63000:II
LINE # 90:FK       LINE # 63001:IA
LINE # 97:JD       LINE # 63002:DP
LINE # 98:MN       LINE # 63003:GJ
LINE # 99:JD       LINE # 63004:FO
LINE # 100:PF      LINE # 63005:OH
LINE # 107:JD      LINE # 63006:JD
LINE # 108:DO      LINE # 63007:MD
LINE # 109:JD      LINE # 63008:FF
LINE # 110:CJ      LINE # 63009:PN
LINE # 117:JD      LINE # 63010:OL
```

**IMPORTANT**
Before typing in an *Ahoy!* program, refer to the first two pages of the program listings section.

```
LINE # 63011:LO
LINE # 63012:NK
LINE # 63013:NC
LINE # 63014:HP
LINE # 63015:LC
LINE # 63016:FN
LINE # 63017:BO
LINE # 63018:AM
LINE # 63019:JH
LINE # 63020:OK
LINE # 63021:HO
LINE # 63022:IK
LINE # 63023:MP
LINE # 63024:BO
LINE # 63025:MI
LINE # 63026:LF
LINE # 63027:DF
LINE # 63028:NA
LINE # 63029:KO
LINE # 63030:JL
LINE # 63031:OP
LINE # 63040:PJ
LINES: 120
```

## BUG REPELLENT LINE CODES FOR VIC 20 CHARACTER SET EXAMPLE

```
LINE # 10:HC
LINE # 15:LG
LINE # 20:CO
LINE # 30:FP
LINE # 40:MI
LINE # 50:AB
LINE # 60:JM
LINE # 70:HK
LINE # 80:HL
LINE # 90:GF
LINE # 100:KJ
LINE # 110:FL
LINE # 120:PF
LINE # 130:OD
LINE # 400:AG
LINE # 460:KK
LINE # 470:MK
LINE # 480:IN
LINE # 490:JO
LINE # 500:MC
LINE # 510:EG
LINE # 520:IM
LINE # 63000:LM
LINE # 63001:OO
LINE # 63002:PB
LINE # 63003:JM
LINE # 63004:JO
LINE # 63005:DE
LINE # 63006:LL
LINE # 63007:FG
LINE # 63008:CB
LINE # 63009:OB
LINE # 63010:FD
LINE # 63011:CE
LINE # 63012:DK
LINE # 63013:AA
LINE # 63014:FP
LINE # 63015:PA
LINE # 63016:GO
LINE # 63017:KH
LINE # 63018:MD
LINE # 63019:CP
LINE # 63020:CP
LINE # 63021:HB
LINE # 63022:KE
LINE # 63023:KH
LINE # 63024:EK
LINE # 63025:HA
LINE # 63026:JF
LINE # 63027:PF
LINE # 63028:DL
LINE # 63029:LL
LINE # 63030:NF
LINE # 63031:BH
LINE # 63040:AN
LINES: 55
```

# SPACE PATROL

FROM PAGE 95

```
•2 SI=54272:FOR A=SI TO SI+24:POK
E A,0:NEXT A
•3 PRINT "[CLEAR]":POKE 53280,0:P
OKE 53281,0:HS=0:X4=0:Y4=0
•4 V=53248:X=60:Y=76:X1=240:Y1=30
:I=0:B=0:F=56095:CO=0:X2=60:Y2=2
10:X3=30:Y3=210
•5 GOSUB 4000:GOSUB 4999
•7 REM ========================
•8 REM ****  TITLE SCREEN  ****
•9 REM ========================
•10 PRINT "[CLEAR]":POKE 53280,0:
POKE 53281,0
•20 FOR A=1 TO 100:X=INT(RND(1)*6
40)+1:C=INT(RND(1)*6)+2
•30 POKE 55456+X,C:POKE 1184+X,46
:NEXT A
•40 PRINT "[HOME][PURPLE][40"[c P
]"]"
•50 PRINT "[UP][18" "][c 7][RVSON
] "
•60 PRINT "[7" "](C) DOUBLE[RVSOF
F] [RVSON]  [RVSOFF] SOFTWARE"
•70 PRINT "[19" "][RVSON]  "
•80 PRINT "[PURPLE][40"[c Y]"]"
•90 PRINTTAB(4)"[RVSON][YELLOW][3
" "][c *][3" "][c *][3" "][c *][
3" "][c *][3" "][c *]"
•100 PRINTTAB(4)"[RVSON][YELLOW]
[RED][3" "][YELLOW] [RED][3" "][
YELLOW] [RED][3" "][YELLOW] [RED
][3" "][YELLOW] [RED][3" "][RVSO
FF][10" "][RED][s Q]"
•110 PRINTTAB(4)"[RVSON][YELLOW]
[RED] [YELLOW] [c *] [RED] [YELL
OW] [RED] [YELLOW] [RED] [YELLOW
] [RED] [YELLOW] [RED] [RVSOFF]
 [RVSON][YELLOW] [RED] [YELLOW][
c *]"
•120 PRINTTAB(4)"[YELLOW][c *][RE
D][RVSON][3" "][YELLOW] [RED][3"
 "][YELLOW] [RED][3" "][YELLOW]
[RED] [RVSOFF]  [RVSON][YELLOW]
[RED]  "
•130 PRINTTAB(4)"[RVSON][YELLOW][
3" "][RED] [YELLOW] [RED] [RVSOF
F]  [RVSON][YELLOW] [RED] [YELLO
W] [RED] [YELLOW] [RED] [YELLOW]
 [c *] [RED] [YELLOW] [c *][RVSO
FF][6" "][GREEN][s Q]"
•140 PRINTTAB(4)"[YELLOW][c *][RV
SON][RED][3" "][YELLOW][RVSOFF][
c *][RVSON][RED] [YELLOW][RVSOFF
]  [c *][RVSON][RED] [YELLOW][RV
SOFF][c *][RVSON][RED] [RVSOFF][
YELLOW][c *][RVSON][RED][3" "][R
VSOFF][YELLOW][c *][RVSON][RED][
3" "]":PRINT
```

```
•150 PRINT TAB(12)"[RVSON][YELLOW
][3" "][c *][3" "][c *][3" "][c
*][3" "][c *][3" "][c *] [c *]"
•160 PRINT TAB(7)"[BLUE][s Q][4"
"][RVSON][YELLOW] [RED][3" "][YE
LLOW] [RED][3" "][YELLOW][RVSOFF
][c *][RVSON][RED][3" "][YELLOW]
 [RED][3" "][YELLOW] [RED][3" "]
[YELLOW] [RED] "
•170 PRINT TAB(12)"[RVSON][YELLOW
] [RED] [YELLOW] [RED] [YELLOW]
[RED] [YELLOW] [RED] [RVSOFF] [R
VSON][YELLOW] [RED] [RVSOFF] [RV
SON][YELLOW] [RED] [YELLOW] [RED
] [YELLOW] [RED] [YELLOW] [RED]
[YELLOW] [RED] "
•180 PRINT TAB(12)"[RVSON][YELLOW
] [RED][3" "][YELLOW] [RED][3" "
][RVSOFF] [RVSON][YELLOW] [RED]
[RVSOFF] [RVSON][YELLOW] [RED]
[RVSOFF][sEP][RVSON][YELLOW] [RE
D] [YELLOW] [RED] [YELLOW] [RED]
 "
•190 PRINT TAB(4)"[c 3][s Q][7" "
][RVSON][YELLOW] [RED] [RVSOFF]
 [RVSON][YELLOW] [RED] [YELLOW]
[RED] [RVSOFF] [RVSON][YELLOW] [
RED] [RVSOFF] [RVSON][YELLOW] [R
ED] [YELLOW] [RED][c *][YELLOW]
[RED] [YELLOW] [RED] [YELLOW] [R
ED] [YELLOW] [c *]"
•200 PRINT TAB(12)"[YELLOW][c *][
RVSON][RED] [RVSOFF]  [YELLOW][c
 *][RVSON][RED] [RVSOFF][YELLOW]
[c *][RVSON][RED] [RVSOFF][YELLO
W] [c *][RVSON][RED] [RVSOFF][YE
LLOW] [c *][RVSON][RED] [RVSOFF]
[YELLOW][c *][RVSON][RED] [RVSOF
F][YELLOW][c *][RVSON][RED][3"
"][RVSOFF][YELLOW][c *][RVSON][R
ED][3" "]"
•210 PRINT:PRINT TAB(8)"[GREEN]DE
SIGNED BY[4"."][c 2]BOB LLORET":
PRINT
•220 PRINT TAB(12)"[YELLOW]PRESS
[c 3]F1 [YELLOW]TO START"
•230 POKE 49522,1:POKE 49523,5:PO
KE 49524,34:POKE 49525,20:POKE 4
9526,24
•235 POKE 53265,PEEK(53265)OR16
•240 POKE 49527,1:SYS 49152:FOR D
=1 TO 90:NEXT D
•270 GET A$:IF A$<>"[F1]" THEN 24
0
•320 POKE 53265,PEEK(53265)AND239
•324 REM ======================
•325 REM ****  PLAYFIELD  ****
•326 REM ======================
•330 PRINT "[CLEAR]":POKE 53280,0
:POKE 53281,0
•340 L$="[HOME][23"[DOWN]"]"
•350 FOR A=1 TO 75:X=INT(RND(1)*7
20)+1:C=INT(RND(1)*6)+2
•360 POKE 55296+X,C:POKE 1024+X,4
6:NEXT A
•365 PRINT LEFT$(L$,19)"[BLUE][40
"[c T]"]"
•370 PRINT LEFT$(L$,22)"[RVSON][W
HITE][9" "][RVSOFF][11" "][CYAN]
[RVSON] SCORE "
•371 POKE V+21,53:SR=51:GOTO 375
•372 POKE V+21,37:SR=35:GOTO 375
•373 POKE V+21,5:SR=3
•375 GOSUB 620
•380 POKE 49522,1:POKE 49523,0:PO
KE 49524,39:POKE 49525,0:POKE 49
526,18
•385 POKE 53265,PEEK(53265)OR16
•390 Y1=30:POKE V+5,Y1:POKE 49527
,1:GOSUB 540
•394 REM ======================
•395 REM ****  MAIN LOOP  ****
•396 REM ======================
•400 SYS 49152
•401 A=INT(RND(0)*3)+601:ON A-601
 GOSUB 601,600
•405 POKE 2042,195:B=B+1:IF B=4 T
HEN GOSUB 610
•410 JV=PEEK(56320):FR=JVAND16:JV
=15-(JVAND15)
•420 IF JV=2 THEN Y=Y+2:IF Y>=180
 THEN Y=180:GOSUB 620
•430 IF FR<>16THENX4=95:Y4=Y-2:PO
KEV+12,X4:POKEV+13,Y4:POKEV+21,S
R+66:GOTO 900
•440 IF JV=1 THEN Y=Y-2:IF Y<=48
THEN Y=48
•445 IF Y1<=50 THEN Y1=50
•446 IF Y1>=174 THEN Y1=174
•450 POKE V+1,Y:POKE V+5,Y1:POKE
2042,194
•460 GOTO 400
•485 REM ========================
==
•490 REM **** SAUCER EXPLOSION **
**
```

**IMPORTANT**

Before typing in an *Ahoy!* program, refer to the first two pages of the program listings section.

```
•495 REM ==========================
==
•500 SYS 49152:POKEV+2,X1:POKEV+3
,Y1:POKE2042,193:FORD=1TO50:NEXT
•501 SYS 49152:POKE SI,0:POKESI+1
,0:POKESI+4,16
•504 FOR A=15 TO 10 STEP -.5
•505 POKESI+24,A:POKESI+4,129:POK
ESI+1,3:POKE SI,5:POKESI+5,11
•506 NEXT A:SYS 49152:HS=HS+100
•510 FOR D=1 TO 5:NEXT:SYS 49152:
POKE V+21,SR-2
•515 PRINT LEFT$(L$,22)TAB(28);"[
CYAN]";HS:SYS 49152
•520 POKE SI,0:POKESI+1,0:POKESI+
4,16:GOTO 380
•525 REM ======================
•530 REM ****  NEW SAUCER  ****
•535 REM ======================
•540 POKE SI+24,2:POKE SI+5,64:PO
KESI+6,130:POKE SI+1,17:POKESI+4
,129:POKESI,37
•550 SYS 49152:C=INT(RND(0)*3)+3:
POKEV+38,C:POKE V+41,1:POKE V+37
,2
•555 POKEV+21,SR+2:POKE 2042,195
•560 SYS 49152:Y1=Y1+3:IF Y1>=100
+(C*2) THEN Y1=100+(C*2):RETURN
•570 POKE 2042,194
•580 POKE V+5,Y1:GOTO 555
•600 SYS 49152:Y1=Y1+7:RETURN
•601 SYS 49152:Y1=Y1-7:RETURN
•607 REM ========================
•608 REM ****  FUEL MONITOR  ****
•609 REM ========================
•610 SYS 49152:F=F-1:POKEF,0:B=0:
IFF=56070THENFORZ=56063TOF-1:POK
EZ,2:NEXT
•612 IF F=56063 THEN H=Y:GOTO 650
•615 RETURN
•620 PRINT LEFT$(L$,20)" [YELLOW]
[RVSON] FUEL [PURPLE][32" "]"
•630 F=56095:B=0:RETURN
•639 REM ==========================
==
•640 REM ** SPACE SHIP EXPLOSION
**
•641 REM ==========================
==
•650 POKE 2040,193:POKEV+21,SR-2:
FORD=1 TO 100:NEXT:POKE V+29,1:P
OKEV+23,1
•660 SYS 49152:POKE SI,0:POKESI+1
,0:POKESI+4,16:CO=CO+1
•670 FOR A=15 TO 8 STEP -.5
•680 POKESI+24,A:POKESI+4,129:POK
ESI+1,3:POKE SI,5:POKESI+5,11
•690 NEXT A:SYS 49152
•700 FOR D=1 TO 45:NEXT:SYS 49152
:POKEV+29,0:POKEV+23,0
•710 POKE SI,0:POKESI+1,0:POKESI+
4,16:POKEV+21,SR-3
•711 IF CO=1 THEN 720
•712 IF CO=2 THEN 800
•713 IF CO=3 THEN 1000
•717 REM ==========================
==
•718 REM ****  2ND SPACE SHIP  **
**
•719 REM ==========================
==
•720 FOR D=1 TO 1000:NEXTD
•730 Y2=Y2-1
•740 POKEV+9,Y2:IF Y2<=76 THEN Y2
=76:Y=Y2:GOTO760
•750 GOTO 730
•760 POKE 2040,192:POKE V+39,5:PO
KEV+21,33:POKE V+1,Y
•770 Y1=30:POKE V+5,Y1:GOTO 372
•797 REM ==========================
==
•798 REM ****  3RD SPACE SHIP  **
**
•799 REM ==========================
==
•800 FOR D=1 TO 1000:NEXT D
•810 X3=X3+1
•820 POKE V+10,X3:IF X3=60  THEN
FOR D=1 TO 500:NEXT D:GOTO 840
•830 GOTO 810
•840 Y3=Y3-1
•850 POKE V+11,Y3:IF Y3=76 THEN Y
=Y3:GOTO 870
•860 GOTO 840
•870 POKE 2040,192:POKE V+39,6:PO
KE V+21,1:POKE V+1,Y
•880 Y1=30:POKE V+5,Y1:GOTO 373
•900 SYS 49152:FOR A=1 TO 25:X4=X
4+5:POKEV+12,X4:NEXT A:SYS 49152
:POKEV+21,SR+2
•910 IF PEEK(V+30)AND64=64 THEN 5
00
•920 GOTO 440
•1000 POKE V+21,0:PRINT"[CLEAR]":
POKE 53280,0:POKE 53281,7
•1010 FOR A=1 TO 8
```

```
•1020 PRINT LEFT$(L$,3)TAB(12)"[2
0" "]"
•1025 FOR D=1 TO 150:NEXT D
•1030 PRINT LEFT$(L$,3)TAB(12)"[P
URPLE]COMPUTER REPORT"
•1035 FOR D=1 TO 150:NEXT D:NEXT
A
•1040 IF HS<=5000 THEN 1060
•1045 PRINT LEFT$(L$,7)TAB(8)"[BL
ACK]S C O R E  ";HS
•1050 MSG$="[5" "]CONGRATULATIONS
[5"."]YOUR MISSION[8" "]WAS A SU
CCESS !!"
•1055 GOTO 1200
•1060 MSG$="[5" "]MICROZ HAS DEFE
ATED YOU AGAIN !!"
•1070 PRINT LEFT$(L$,7)TAB(8)"[BL
ACK]S C O R E  ";HS
•1200 FOR A=1 TO LEN(MSG$)
•1210 PRINT "[PURPLE]";LEFT$(L$,1
0)LEFT$(MSG$,A)
•1220 FOR D=1 TO 150:NEXT D:NEXT
A
•1230 END
•3980 REM =======================
•3990 REM ****  SPRITE DATA  ****
•3995 REM =======================
•4000 FOR S1=12288 TO 12350:READ
D1:POKES1,D1:NEXT S1
•4005 DATA 0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
•4010 DATA 240,56,0,124,68,0,63,2
55,224,63,255,252,63,255,255
•4015 DATA 0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0
•4020 POKE V+39,2:POKE 2040,192:P
OKE V,X:POKE V+1,Y
•4025 FOR S2=12352 TO 12414:READ
D2:POKE S2,D2:NEXT
•4030 DATA 128,0,0,64,0,24,104,0,
57,52,48,114,26,56,100,14,184,20
0,2
•4035 DATA 185,0,0,178,0,252,176,
58,240,32,252,1,1,255,126,0,0,16
,132
•4040 DATA 15,35,4,64,70,6,32,140
,34,56,24,182,92,51,182,110,101,
172
•4045 DATA 82,73,176,0,0,128,0
•4050 POKE 2041,193:POKE V+40,2
•4060 FOR S3=12416 TO 12478:READ
CH:POKE S3,CH:NEXT
•4070 DATA 0,0,0,0,0,0,0,0,0,0,0,
0,0,255,0,3,255,192,15,255,240,6
3,255,252
•4080 DATA 255,255,255,85,85,85,1
87,187,187,85,85,85,255,255,255
•4090 DATA 63,255,252,15,255,240,
3,255,192,0,255,0,0,0,0,0,0,0,0,
0,0,0,0,0
•4100 POKE 2042,194:POKEV+28,4
•4110 POKE V+4,X1:POKE V+5,Y1
•4120 FOR S4=12480 TO 12542:READ
D4:POKE S4,D4:NEXT
•4130 DATA 0,0,0,0,0,0,0,0,0,0,0,
0,0,255,0,3,255,192,15,255,240,6
3,255,252
•4140 DATA 255,255,255,85,85,85,2
38,238,238,85,85,85,255,255,255
•4150 DATA 63,255,252,15,255,240,
3,255,192,0,255,0,0,0,0,0,0,0,0,
0,0,0,0,0
•4160 POKE 2043,195:POKE V+41,1:P
OKEV+28,4:POKE V+37,5:POKEV+38,4
•4170 FOR S5=12544 TO 12606:READ
D5:POKES5,D5:NEXT S5
•4180 DATA 0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
•4190 DATA 240,56,0,124,68,0,63,2
55,224,63,255,252,63,255,255
•4200 DATA 0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0
•4210 POKE2044,196:POKEV+43,5:POK
EV+8,X2:POKEV+9,Y2
•4220 FOR S6=12608 TO 12670:READ
D6:POKES6,D6:NEXT S6
•4230 DATA 0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
•4240 DATA 240,56,0,124,68,0,63,2
55,224,63,255,252,63,255,255
•4250 DATA 0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0
•4260 POKE2045,197:POKEV+44,6:POK
EV+10,X3:POKEV+11,Y3
•4270 FOR S7=12672 TO 12734:READ
D7:POKE S7,D7:NEXT S7
•4280 DATA 0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0
•4290 DATA 0,0,0,0,0,0,0,0,0,255,
255,255,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0
•4300 DATA 0,0,0,0,0,0
•4310 POKE 2046,198:POKEV+45,2
•4320 RETURN
•4985 REM =======================
```

## IMPORTANT

Before typing in an *Ahoy!* program, refer to the first two pages of the program listings section.

```
===
•4990 REM ** ML SCREEN SUBROUTINE
 **
•4995 REM ========================
===
•4999 FOR I=1 TO 4:READ A:NEXT:RE
AD LO,HI:FORI=LO TO HI:READX:POK
E I,X:NEXT
•5000 DATA 11507,12573,12522,1100
1
•5005 DATA 49152, 49528
•5006 RETURN
•5010 DATA 174,114,193,224,3,144,
3,76,117,192,188,114,193,140,121
,193,174
•5020 DATA 118,193,232,202,32,30,
193,172,121,193,173,119,193,201,
2,208,10
•5030 DATA 169,32,72,173,33,208,7
2,76,50,192,177,90,72,177,92,72,
204
•5040 DATA 116,193,240,20,200,177
,90,72,177,92,136,145,92,104,145
,90,200
•5050 DATA 204,116,193,208,238,24
0,18,136,177,90,72,177,92,200,14
5,92,104
•5060 DATA 145,90,136,204,115,193
,208,238,173,119,193,201,0,208,5
,104,104
•5070 DATA 76,111,192,104,145,92,
104,145,90,236,117,193,208,160,9
6,172,116
•5080 DATA 193,200,189,114,193,17
0,32,30,193,173,120,193,201,2,20
8,19,136
•5090 DATA 169,32,153,122,193,173
,33,208,153,162,193,204,115,193,
208,239,240
•5100 DATA 16,136,177,90,153,122,
193,177,92,153,162,193,204,115,1
93,208,240
•5110 DATA 236,117,193,240,37,202
,32,30,193,172,116,193,200,136,1
77,90,72
•5120 DATA 177,92,32,48,193,145,9
2,104,145,90,32,56,193,204,115,1
93,208
•5130 DATA 234,236,117,193,208,22
1,240,46,202,206,118,193,232,32,
30,193,172
•5140 DATA 116,193,200,136,32,48,
193,177,90,72,177,92,32,56,193,1
45,92
•5150 DATA 104,145,90,204,115,193
,208,234,236,118,193,208,221,238
,118,193,232
•5160 DATA 32,30,193,173,120,193,
201,0,240,20,172,115,193,136,200
,185,162
•5170 DATA 193,145,92,185,122,193
,145,90,204,116,193,208,240,96,1
89,89,193
•5180 DATA 133,91,24,105,212,133,
93,189,64,193,133,90,133,92,96,7
2,152
•5190 DATA 24,105,40,168,104,96,7
2,152,56,233,40,168,104,96,0,40,
80
•5200 DATA 120,160,200,240,24,64,
104,144,184,224,8,48,88,128,168,
208,248
•5210 DATA 32,72,112,152,192,4,4,
4,4,4,4,4,5,5,5,5,5
•5220 DATA 5,6,6,6,6,6,6,6,7,7,7,
7,7,3,0,4,0
•5230 DATA 4,1,1
```

### BUG REPELLENT LINE CODES FOR SPACE PATROL

```
LINE # 2:DM        LINE # 200:AO
LINE # 3:JB        LINE # 210:OM
LINE # 4:GH        LINE # 220:JN
LINE # 5:OI        LINE # 230:FL
LINE # 7:HL        LINE # 235:AN
LINE # 8:BG        LINE # 240:AH
LINE # 9:HL        LINE # 270:MN
LINE # 10:OD       LINE # 320:FC
LINE # 20:KK       LINE # 324:KC
LINE # 30:CE       LINE # 325:LL
LINE # 40:OL       LINE # 326:KC
LINE # 50:IM       LINE # 330:OD
LINE # 60:ED       LINE # 340:LK
LINE # 70:IP       LINE # 350:JC
LINE # 80:NM       LINE # 360:DH
LINE # 90:NN       LINE # 365:IN
LINE # 100:AD      LINE # 370:OE
LINE # 110:EO      LINE # 371:KE
LINE # 120:BN      LINE # 372:PM
LINE # 130:DA      LINE # 373:CA
LINE # 140:JB      LINE # 375:CE
LINE # 150:MM      LINE # 380:CF
LINE # 160:BP      LINE # 385:AN
LINE # 170:EP      LINE # 390:EH
LINE # 180:KD      LINE # 394:KC
LINE # 190:OM      LINE # 395:FO
```

## IMPORTANT

Before typing in an *Ahoy!* program, refer to the first two pages of the program listings section.


```
LINE # 396:KC
LINE # 400:KF
LINE # 401:KH
LINE # 405:PL
LINE # 410:NC
LINE # 420:PG
LINE # 430:NG
LINE # 440:IO
LINE # 445:CH
LINE # 446:KA
LINE # 450:OP
LINE # 460:CA
LINE # 485:ME
LINE # 490:CA
LINE # 495:ME
LINE # 500:PK
LINE # 501:DC
LINE # 504:LB
LINE # 505:DK
LINE # 506:PI
LINE # 510:AO
LINE # 515:MM
LINE # 520:JI
LINE # 525:PM
LINE # 530:IK
LINE # 535:PM
LINE # 540:II
LINE # 550:CA
LINE # 555:AM
LINE # 560:HA
LINE # 570:ED
LINE # 580:MN
LINE # 600:DD
LINE # 601:DC
LINE # 607:HL
LINE # 608:HK
LINE # 609:HL
LINE # 610:OK
LINE # 612:PF
LINE # 615:IM
LINE # 620:HE
LINE # 630:DF
LINE # 639:ME
LINE # 640:DN
LINE # 641:ME
LINE # 650:GC
LINE # 660:PB
LINE # 670:GM
LINE # 680:DK
LINE # 690:LE
LINE # 700:PH
LINE # 710:JK
LINE # 711:FK
LINE # 712:EE
LINE # 713:HG
LINE # 717:ME
LINE # 718:PD
LINE # 719:ME
LINE # 720:AG
LINE # 730:JD
LINE # 740:AF
LINE # 750:CG
LINE # 760:JG
LINE # 770:NK
LINE # 797:ME
LINE # 798:AA
LINE # 799:ME
LINE # 800:AG
LINE # 810:LG
LINE # 820:ND
LINE # 830:CN
LINE # 840:KL
LINE # 850:AF
LINE # 860:CI
LINE # 870:MN
LINE # 880:NN
LINE # 900:CK
LINE # 910:KK
LINE # 920:CE
LINE # 1000:EA
LINE # 1010:IL
LINE # 1020:ME
LINE # 1025:OF
LINE # 1030:AK
LINE # 1035:OJ
LINE # 1040:OO
LINE # 1045:LG
LINE # 1050:OO
LINE # 1055:FE
LINE # 1060:NN
LINE # 1070:LG
LINE # 1200:ID
LINE # 1210:EJ
LINE # 1220:OJ
LINE # 1230:IC
LINE # 3980:CD
LINE # 3990:LF
LINE # 3995:CD
LINE # 4000:LJ
LINE # 4005:ND
LINE # 4010:JG
LINE # 4015:DH
LINE # 4020:CG
LINE # 4025:EP
LINE # 4030:NI
LINE # 4035:JA
LINE # 4040:ID
LINE # 4045:BD
LINE # 4050:NI
LINE # 4060:FE
LINE # 4070:CP
LINE # 4080:OP
LINE # 4090:IJ
LINE # 4100:CK
LINE # 4110:HF
LINE # 4120:IL
LINE # 4130:CP
LINE # 4140:NE
LINE # 4150:IJ
LINE # 4160:LL
LINE # 4170:FA
LINE # 4180:ND
LINE # 4190:JG
LINE # 4200:DH
LINE # 4210:OA
LINE # 4220:KN
LINE # 4230:ND
LINE # 4240:JG
LINE # 4250:DH
LINE # 4260:AF
LINE # 4270:BG
LINE # 4280:FC
LINE # 4290:IE
LINE # 4300:JC
LINE # 4310:PP
LINE # 4320:IM
LINE # 4985:ME
LINE # 4990:OH
LINE # 4995:ME
LINE # 4999:BN
LINE # 5000:MD
LINE # 5005:OB
LINE # 5006:IM
LINE # 5010:GI
LINE # 5020:EL
LINE # 5030:CN
LINE # 5040:CI
LINE # 5050:AK
LINE # 5060:KD
LINE # 5070:OJ
LINE # 5080:OP
LINE # 5090:KB
LINE # 5100:CE
LINE # 5110:MG
LINE # 5120:HJ
LINE # 5130:GF
LINE # 5140:IC
LINE # 5150:AL
LINE # 5160:DK
LINE # 5170:EP
LINE # 5180:IA
LINE # 5190:LL
LINE # 5200:LK
LINE # 5210:JG
LINE # 5220:LE
LINE # 5230:HP
LINES: 218
```

# PRINTER INTERFACING REVISITED

*Continued from page 21*

will set the print head to the 256+15=271 dot position.

## *THE REPEAT GRAPHICS COMMAND*

The 1525 can be instructed to repeat a graphic dot pattern up to 255 times. This can be very useful for generating graphs and bar charts. The CHR$(26) code is followed by a single byte which specifies the number of repetitions. The pattern to be repeated is sent as a single byte immediately afterward. The following program demonstrates the proper application of the dot position and graphic repeat commands.

```
10 OPEN4,4
20 PRINT#4,CHR$(8);
30 PRINT#4,CHR$(27)CHR$(16)CHR$(0
)CHR$(100);
40 PRINT#4,CHR$(26)CHR$(50)CHR$(2
55)
50 PRINT#4,CHR$(27)CHR$(16)CHR$(1
)CHR$(200);
60 PRINT#4,CHR$(26)CHR$(50)CHR$25
5)
70 PRINT#4,CHR$(15)
80 CLOSE4
```

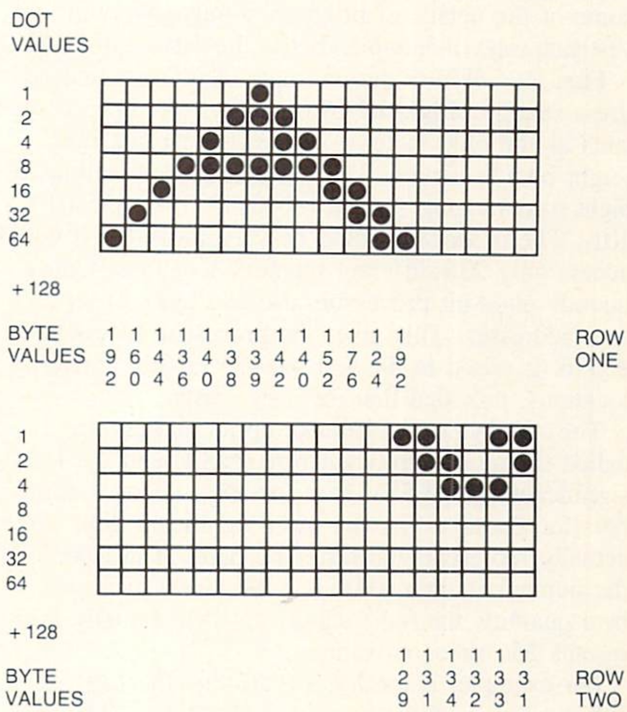
Line 30 moves the printhead to the 100th dot position. Line 40 prints a solid bar for the next 50 dot positions. Line 50 moves the printhead to the 1*256+200=456th dot position. This forces the 50 dot bar in line 60 to overflow to the next printed line. Remember, a line on the 1525 is 480 dots long.

## *PRINTING HIGH RESOLUTION GRAPHICS*

In text mode, the 1525 prints six lines to the inch. Each character is made up of a matrix which is six dots wide and seven dots high. When high resolution graphics mode is selected by the CHR$(8) code, the line spacing is changed to nine lines per inch. In this mode, successive rows are printed adjacent to each other. Any data following the CHR$(8) code will be interpreted as printable graphic data if the byte value is greater than 127. Byte values less than 128 are either ignored or are one of the command codes described above. The following program demonstrates the graphics mode by printing the first letter of the *Ahoy!* logo.

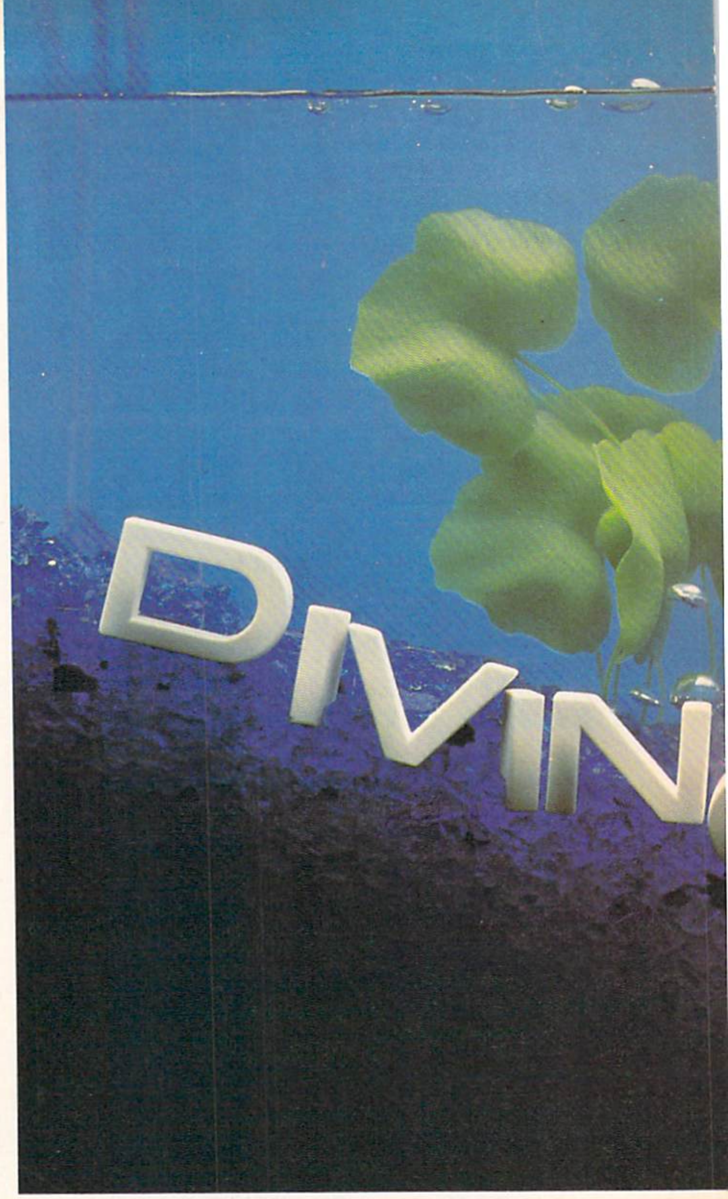
```
10 DATA192,160,144,136,140,138
20 DATA139,142,140,152,176,224
30 DATA192
40 DATA129,131,134,132,133,131
50 FOR I=0 TO 12
60 READ R1
70 R1$=R1$+CHR$(R1)
80 NEXT I
90 FOR I=0 TO 5
100 READ R2
110 R2$=R2$+CHR$(R2)
120 NEXT I
130 OPEN4,4
140 PRINT#4,CHR$(8)R1$
150 PRINT#4,CHR$(27)CHR$(16)CHR$(
0)CHR$(12)R2$CHR$(15)
160 CLOSE 4
```

The data in lines 10 through 40 make up the actual image. The following figure shows how it is derived.




Each pass of the printhead, in graphic mode, can print a swath exactly seven dots high. Each vertical seven-dot column is represented by a single byte of data. The value of this byte is dependent on which dots are to be printed. Referring to the figure, the top dot has a value of one and the bottom dot has a value of 64. The value of 128 is added in to tell the printer that this is graphic data. For a more detailed example of high resolution printer graphics, see *Lower Case Descenders with the VIC1515/1525 Printers* in the April issue of *Ahoy!*

The examples presented above require either a Commodore 1525 printer or a dot matrix printer with a graphic interface. In February we will be taking a close look at the latest offerings in the Commodore printer interface market. □




# Exploring the

Do you ever wonder what is going on inside your computer when it executes a BASIC program? In previous columns we have seen how numbers and arrays are stored. We have also taken a quick look at the way the program itself is stored. This month we will see some of the details of program storage. We will investigate ways of snooping below the surface of BASIC.

First let's discuss the concepts of pointers and address storage in BASIC. An eight-bit microprocessor such as the 6510 in the C-64 reads data one byte (eight bits) at a time. There are 256 combinations of eight bits, ranging from 0 (0000 0000) to 255 (1111 1111). The processor would be very limited if it could access only 256 different memory locations. Consequently eight-bit processors use two bytes to represent addresses. This gives the processor access to 65536 (2 raised to the 16th power) possible memory locations, provided that memory exists.

The two bytes which make up an address are called the most significant byte (MSB) and the least significant byte (LSB). In a two-digit decimal number (for example 37), the most significant digit (3) actually represents ten times its value. The value of the number is then 10*3 + 7. Similarly in a two-byte quantity, the most significant byte actually represents 256 times its value.

For example, if the MSB is 30 and the LSB is 135, the value of the quantity is 30*256 + 135 or 7815. The smallest two byte quantity is MSB=0, LSB=0 which has a value of 0. The largest two byte quantity is MSB=255, LSB=255 which has a value of 65535 (you should calculate it to make sure).

There is one twist. When a two byte quantity is stored in memory, it is stored (contrary to common sense) with the LSB first and the MSB second. If the computer reads two bytes 100 and 10 when it is expecting an address, it determines that the address is actually 100 + 256*10 or 2660.

The BASIC interpreter is a program in ROM which interprets and executes the instructions which you type into RAM as your BASIC program. BASIC looks into specific RAM locations to find out information about your program. For example, RAM locations at addresses 43 and 44 contain a two byte "pointer" to the start of your BASIC program in RAM.

What is a pointer? It is nothing more than a two byte address. Pointers are stored in standard LSB, MSB sequence. If for example you type:

```
PRINT PEEK(43) PEEK(44)
```

in direct mode, you will see the LSB and the MSB of the address where your BASIC program begins (or will begin). If your results are 1 and 8, that means that your program starts in RAM at address 2049. When we refer to the value of the pointer at location 43, we actually mean the value of the byte at address 43 added to 256 times the value of the byte at address 44. The following program defines a function FNPT(M) in line 10 which evaluates the pointer at address M.

```
1 REM...BASIC-LOOK...
10 DEFFNPT(M)=PEEK(M)+256*PEEK(M+
1)
20 M1=FNPT(43):M2=FNPT(45)-1
```




Photo by Ross M. Horowitz/Concept by Case

# Inner Structure of a BASIC Program

By Dale Rupert

```
30 FORMEM=M1TOM2
40 P=PEEK(MEM)
50 PRINTMEM;P;
60 IFP<31ORP>128THENPRINT:GOTO80
70 PRINTCHR$(P)
80 NEXT
```

If you type it exactly as shown including line 1 and without spaces, you should see the following results. The program starts at address 2049 and ends at address 2206. This program displays each address and the value of the byte stored in each address of this program. Also for bytes which have a character equivalent, that character is displayed.

Notice that the pointer at address 45 gives the address of memory just beyond the BASIC program. That is where variables and their values are stored. You may also notice that there aren't any BASIC statements as there are when you list the program. In fact about the only recognizable items are variable names and assignment values as well as the words in the REM statement.

The BASIC interpreter converts your program's text into a tokenized format for storage. Each statement has its own one byte code or token. For example, after you run the previous program, the NEXT statement in line 80 will be near the bottom of your screen, disguised as code 130 in address 2203. Further up you should see the PRINT and CHR$ keywords stored as 153 and 199 respectively.

You may run the following program to determine the tokens for all BASIC statements and keywords.

```
1 REM S+0..,1..,2..,3..,4..,5..,6
..,7..,8..,9..
2 S=120 : REM > STARTING CODE
10 DEF FNPT(M)=PEEK(M)+256*PEEK(M
+1)
```

# Just when you've escaped Castle Wolfenstein™ MUSE introduces another game to capture your imagination.




## Announcing Beyond Castle Wolfenstein™

Creating the original Castle Wolfenstein™ was indeed a challenge. Developing a sequel to our award winning top seller was even tougher. But we didn't surrender. Now there's Beyond Castle Wolfenstein™—an action/adventure classic in the Castle Wolfenstein™ tradition.

So if you take your strategy games seriously, accept our challenge to play Beyond Castle Wolfenstein™—the game you've been waiting for.

MUSE Software also offers an exceptional selection of quality, critically acclaimed home management, personal productivity, educational and recreational software.

There's Super-Text® Professional word processor—the highly recognized top seller that will make all your writing easier. Elementary Math™ and Antonym Antics™—two great teaching programs that assure your child a head start in learning. And Space Taxi™—the fast-paced, action game of 25 full color screens that will test even the best of drivers.

As always, you can expect the exceptional . . . from MUSE Software.













**MUSE®**
**SOFTWARE**


347 N. Charles Street
Baltimore, Maryland 21201
(301) 659-7212


Super-Text® Professional, Rescue Squad™, Space Taxi™ and Castle Wolfenstein™ are available for the Commodore 64.

```
•20 M1=FNPT(43):M0=M1+10
•30 FOR N=0 TO 9
•35 IF (S+N=204) OR (S+N>255) THE
N POKE M0,32 : GOTO 50
•40 POKE M0,S+N
•50 M0=M0+4
•60 NEXT
•70 LIST 1-2
```

Type line 1 exactly as shown. BASIC tokens have values starting at 128. This program POKEs ten consecutive values into the RAM locations coinciding with the REM statement in line 1. The rather obscure display when you run this program represents a listing of keywords and their corresponding code number. The starting value S is listed in line 2. The first eight symbols shown are regular graphics characters. Item 8 is the keyword END, and item 9 is the keyword FOR. This says that END has a token value of 128, and FOR's value is 129.

Now change the 120 in line 2 to 130 and run the program again. This time the keywords from NEXT to IF are displayed, corresponding to tokens 130 through 139. You may continue changing line 2 in steps of ten up to 250 (you can't POKE a value larger than 255). You may modify this program to send the results to a printer if you prefer.

In this program, M1 stores the starting address of program storage. Sequential values are POKEd into memory starting at address M1 + 10. Why M1 + 10? To answer that, we must look more closely at the way our programs are stored in memory.

## *BASIC STORAGE*

The diagram on page 91 (Figure 1) shows the structure of a BASIC program. Each square represents a memory cell. The addresses of the cells are identified. The contents of the cells depend upon which particular program is in memory.

The pointer at address 43 tells where the program begins. The first two bytes of the program are another pointer. This pointer shows where the second line of the program begins. The format is the same for each program line. The first two bytes of each line point to the next program line. The second pair of bytes store the line number of the line (in LSB, MSB form). The program text is stored in the remaining bytes of each line. The end of each line is indicated by one byte of zero.

The last program line also starts with a pointer. This pointer points to two consecutive bytes of zeroes. That is the signal for the computer to identify the end of the program. If you rerun the first program, you will see three consecutive bytes of zero, one corresponding to the end of line 80 and two for the end of the program. Notice that the program storage is in contiguous memory locations. There is no gap in memory between the end of one line and the beginning of the next.

Now we can go back and see why the previous program works. M1 is the starting address of the program. The pointer to the second line is stored in locations M1 and M1+1. The line number of the first line is stored in locations M1+2 and M1+3. The text of the first line begins at location M1+4. The token for the REM statement is in M1+4. The space after the REM is in M1+5. The following diagram shows how the first bytes of the text in the first line are stored:

| M1 | +4 | +5 | +6 | +7 | +8 | +9 | +10 | +11 | +12 | +13 | +14 | +15 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | REM | (space) | S | + | 0 | . | . | . | 1 | . | . | . |

When we POKE a value into M1+10, we are replacing the second period following the "0" with that value. The next value we POKE goes into M1+14 which replaces the second period after the "1". Keep in mind that, even though the listing of the line shows all the characters of the keyword, each keyword token is stored in memory as a single byte.

The way that pointers work seems a bit magical. When we evaluate the pointer at address 43, that is FNPT(43), we get an address, say A1. Evaluating the pointer at address A1 gives another address A2. If we wanted to find the starting address of the third line of the program, we could evaluate

```
A3 = FNPT(FNPT(FNPT(43)))
```

where

```
FNPT(M) = PEEK(M) + 256*PEEK(M+1)
```

Similarly we could evaluate the pointer of the pointer of the pointer. . .until we found a value of zero, indicating the end of the program.

Another way to find the end of the program text that we have typed is to use the pointer at address 45. This pointer tells the start of numeric variable storage. The end of the program text (not including the zero at the end of the last line) is always four bytes before that. Consequently, we may find the end of our text from the expression

```
FNPT(45)-4
```

The following program analyzes itself as an example of how we may investigate a program *in situ*.

```
•2000 :REM...BASIC LINE LOCATER..
.
•2005 DEF FNPT(X)=PEEK(X)+256*PEE
(X+1)
2010 GOSUB 2200 :REM COUNT # OF L
INES
2015 PRINT"THERE ARE"CT"PRGM. LIN
ES"
2020 PRINT:INPUT"WHICH PRGM. LINE
";N
2025 IF N>CT OR N<1 THEN 2015
2030 GOSUB 2500 : REM GET LINE AD
DRESS
2040 PRINT"FOR PRGM. LINE"N
2050 PRINT" LINE # = "FNPT(A+2)
2060 PRINT" TEXT STARTS AT ADDR =
 "A+4
2070 PRINT" TEXT ENDS AT ADDR   =
 "NL-2
2100 GOTO 2020
2199 :
2200 :REM....COUNT # OF LINES....
.
2201 :
2210 A=FNPT(43)
2220 CT=0
2230 IF FNPT(A)=0 THEN2270
2240 CT=CT+1
2250 A=FNPT(A)
2260 GOTO 2230
2270 RETURN
2499 :
2500 REM...GET ADDR OF NTH LINE..
2501 :
2510 A=FNPT(43) : REM FIRST LINE
ADDR
2520 IF N<2 THEN 2560
2530 FOR QQ=2 TO N
2540 A=FNPT(A)
2550 NEXT QQ
2560 NL=FNPT(A) :REM START OF NEX
T LINE
2570 RETURN
```

The subroutine at line 2200 shows how a program might count its number of lines. The subroutine at 2500 steps through the specified number of lines (by stepping from pointer to pointer). The value of A is the address of the start of the Nth line. The line number of the Nth line is at locations (A+2) and (A+3) and is evaluated just as a pointer is. The text in the Nth line starts at address (A+4). NL is the pointer to the next line as defined in line 2560. Consequently the end of the text in this line is at address (NL−2).

Instead of searching for the Nth line in a program, we might prefer to search for the line with a given line number. We now know where to find the line number of each line. Since it is in "pointer format," we must first break the desired line number into LSB, MSB form. The following program does that.

```
•1 :REM CONVERT # TO LSB,MSB FORM
.
•2 DEF FNMS(X)=INT(X/256)
•3 DEF FNLS(X)=X-256*INT(X/256)
•4 INPUT"CONVERT WHAT NUMBER";X
•5 PRINT"LSB, MSB:"FNLS(X);FNMS(X
)
•6 GOTO 4
```

Line numbers may range from 0 to 63999 (somewhat less than the 65536 possible values permissible with two bytes). The LSB and MSB values for 63999 are 255, 249. You may enter numbers as large as you like into this program, but the MSB will be larger than 255 for numbers greater than 65535.

This program and the subroutines from the previous programs are purely illustrative since they only work on themselves. To really utilize these ideas, we could put these routines at the end of some other program we wish to analyze. Now that we know how pointers work, let's look at a better way of combining these, or any other utilities, with other programs.

### *THE GREAT MERGER*

When we load a program into memory, BASIC checks the pointer at address 43 (remember it?) to find where the program should start. After it is loaded, the pointer at address 45 is updated to show where the program ends, actually where numeric variable storage begins, as we have already discussed.

It is an easy task for us to change those pointers and thereby to join two programs into one. The procedure is this:

1. In direct mode, type A = PEEK(43) : B = PEEK(44) : PRINT A,B
2. Load the first program (the one with smaller line numbers) into memory as you normally would.
3. Type C = PEEK(45) : D = PEEK(46) : PRINT C,D

4a. If C is not 0 or 1, type POKE 43,C−2 : POKE 44,D

OR 4b. If C=0 or 1, type POKE 43,C+254 : POKE 44,D−1

5. Load the second program into memory.
6. In direct mode, type POKE 43,[A] : POKE 44, [B] where [A] and [B] are the values from step one, normally 1 and 8.

Step 1 determines the starting address of the first program. C and D are pointers to two bytes beyond the end of that program once it is loaded. Step 4 sets the end of the first program as the start of the second program. You do either 4a or 4b, not both. Step 4b is a subtraction with a borrow from the MSB. Once the second program is loaded into memory, BASIC properly sets the pointer at 45. If we listed the program now, only the second program would be shown because of the pointer value at 43. By restoring the pointer at 43 back to its initial value, we have joined the two programs, and we may treat them as one.

The whole process may be repeated if we want to add another program. If successive programs do not have sequentially increasing line numbers, the programs will load anyway, but BASIC will have trouble with the lines out of sequence. A simple way around this is to list the program and press <Return> for each program line in the second program which should be contained within the first program. You will then have duplicate lines but the second line in each case will be ignored.

A better procedure is to renumber your programs before merging them so that they will be in sequential order.

Having the ability to merge program segments together should encourage you to write some programs in a modular format and create others from the basic modules.

## *CROSS REFERENCE*

The final program this month may be merged to another program you wish to analyze for program flow. This program prints out a list of each line number which branches to other lines. It also gives a list of each line number and the lines which branch to it. See the listing on page 73.

The program uses the techniques we have just discussed. The array LN( ) stores information about each program line in the following format:

LN(A,0) is the line number of the Ath line of the program.

LN(A,1) is the count of how many branches are made by this line.

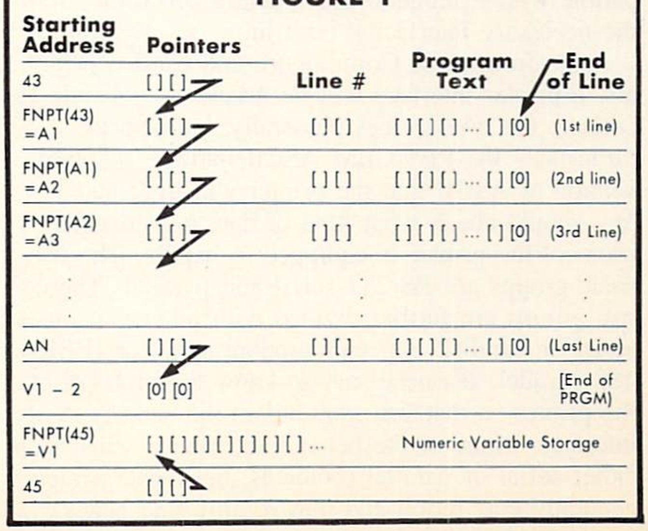
LN(A,2) through LN(A,. . .) are the line numbers

*Continued on page 98*

## READER SERVICE INDEX

| Page No. | Company | Svc. No. |
|---|---|---|
| 15 | Commodore Business Machines | 1 |
| 15 | Star Micronics | 2 |
| 16 | Cardco, Inc. | 3 |
| 16 | Cardco, Inc. | 4 |
| 5 | Datasoft | 5 |
| 7 | Datasoft | 6 |
| 37 | Mirage Concepts, Inc. | 7 |
| 38 | Cardco, Inc. | 8 |
| 39 | Loadstar | 9 |
| 6 | Waveform Corporation | 10 |
| 6 | Orange Micro Inc. | 11 |
| 9 | EnTech Software | 12 |
| 96 | Activision | 13 |
| 14 | Creative Software | 14 |
| 14 | Victory Software | 15 |
| 14 | Victory Software | 16 |
| 50-51 | Sight & Sound Music Software | 17 |
| 96 | HesWare | 18 |
| 96 | HesWare | 19 |
| 22-33 | Protecto Enterprizes | 20-31 |
| 49 | CVC Online | 32 |
| 12 | 3R Import and Export Corp. | 33 |
| 41 | Ergo Systems | 34 |
| 9 | Cheatsheet Products | 35 |
| 11 | Eng Mfg., Inc. | 36 |
| 54 | Parsec Research | 37 |
| 14 | Starpoint Software | 38 |
| 60 | PLI Micro | 39 |
| 12 | Susie Software | 41 |
| 13 | Mimic Systems Inc. | 42 |
| C-2 | Cardco, Inc. | 43 |
| 57 | Datamost | 44 |
| 18 | Indus Systems | 45 |
| 35 | HesWare | 46 |
| 35 | Professional Software | 47 |
| 88 | Muse Software | 48 |
| C-3 | Activision | 49 |
| 42 | Melodian | 50 |
| 34 | MMG Microsoftware | 51 |
| 40 | GOSUB of Slidel, Inc. | 52 |
| 36 | PSIDAC | 53 |
| C-4 | Koala Technologies Corporation | 54 |
| 8 | subLOGIC Corporation | 55 |
| 97 | Activision | 56 |

FIGURE 1



# PRIMER ON PRINTERS

*Continued from page 17*

often have optional tractor feed attachments available as well. Tractor feed assures smooth, even feed of paper through the printer even at high speeds without the paper shifting.

## *NOISE*

Impact printers, both dot matrix and letter quality, are noisy machines. As a general rule, the letter quality variety is somewhat noisier than the dot matrix group, but both types do make a considerable amount of noise. Using a cushion under the printer will lessen the racket somewhat, and there are many good printer mats available commercially. Most are made up of a spongy rubber, with some having a porous "waffle" texture which not only cuts down on the inherent mechanical noise of the printer, but lessens the amount of vibration, which also contributes to the noise level. It's a good idea to take a look at these printer mats when you purchase your printer, but there are two things in particular to look for: 1) make sure that the mat is large enough to accommodate the printer properly, and 2) be very sure that the mat will not impede the flow of air into any vents on the bottom of the printer. This is important, because if these vents are blocked, the workings will get too hot, and printer failure will result.

## *COMPATIBILITY*

Commodore computers transmit PET/CBM ASCII codes through their serial ports, which is a modified version of standard ASCII. Commodore printers, as one would expect, understand this modified code dialect perfectly, and thus will operate properly without the use of an interface.

Cardco, Inc. offers three letter quality printers, the LQ-1, LQ-2, and LQ-3, which are also directly compatible with Commodore computers, since they have the necessary interfacing built in.

Aside from these Commodore and Cardco printers, a printer interface will be necessary not only to connect the two devices physically, but also to "translate" the PET/CBM ASCII into the standard version of ASCII that the printer will understand. You should check what kind of data transmission protocol the printer is equipped to handle. The two basic groups are RS-232 serial and parallel. These two groups are further divided with other versions of serial or parallel, e.g. Commodore serial or IEEE-488 parallel. It's necessary to know the protocol of the printer so that you may obtain the right type of interface. Many of the better line printers will accept either serial or parallel protocols, but these printers generally cost more, and they're all-round better units. For a more detailed explanation of interfaces, as well as more information on these amazing devices, see Morton Kevelson's article on page 19.

## *PRINT VARIETY*

The ability to alter or vary the size and style of print is a feature that virtually all printers are capable of to some extent, more or less depending on the individual printer as well as whether it's dot matrix or letter quality.

Most letter quality printers offer optional print wheels or print elements to afford you a change in what the characters look like, as well as what special characters they may contain. For example, a "scientific" print wheel layout might contain such characters as pi, degree, and other specialized scientific or mathematical characters. The standard "courier" or "legal" varieties of print wheels contain pretty much the standard assortment of characters found on any good typewriter, while offering such niceties as a bullet in place of a math symbol. While the variety of print wheels for any particular printer is somewhat limited, there is still an adequate variety of layouts for most purposes. Most letter quality printers allow double- or triple-striking characters to create a pseudo-bold face, and by substituting a print wheel of a different pitch, the print can be changed from pica to elite or vice versa.

Dot matrix printers offer all kinds of goodies in the way of altering print formats. Many are capable of italicizing, subscripts, superscripts, boldface, underlining, expanded, or condensed print, in addition to the ability to print out the graphics characters of your computer. Some of the better dot matrix units also have alternate or auxiliary character sets on ROM within the machine, such as the kata-kana character set, accent marks such as umlauts and circumflex characters, etc. Additionally, many of these units allow you the option to load into memory your own custom character set for further specialized printing needs. As with all of the other added features, this capability is usually reflected in the price of the printer.

## *INTERNAL BUFFER MEMORY*

The internal memory, or buffer, of the printer, no matter which genre you choose, greatly influences the speed of the unit. A printer with a small buffer, say 64 characters, can store and print only one line of characters at a time, whereas a printer with a 2K buffer can store and print an entire page or document at a time. This is an important feature, since the more characters the printer can hold in RAM, the quicker the computer is freed up from playing nursemaid to the printer, thus allowing you to use it for other tasks or operations while the printer does its thing. On the surface it doesn't seem like a very big point, until you have a rather long document or

listing to print out, and constantly waiting for the printer to catch up with the computer becomes a time-consuming ordeal. Many of the printers offer the option of installing an additional RAM chip or chips by the user if and when the need for more buffer memory arises. These printers offer the best value in this respect, since you're only paying for the buffer memory you actually need, and you only pay for it when you need it!

## *COMPARISON CHART*

Below is a chart of the most popular printers, their type, features and manufacturers' suggested list prices. It is by no means complete, but it does represent an excellent cross-section of the more popular brands in different price categories. All of these units were selected for our comparison because they offer the best variety in their particular groups and price ranges. □

## PRINTER FEATURE COMPARISON CHART

| Mfr./Model | Image Type | Speed | Paper Feed | Paper Width (max.) | Buffer Memory | Compatibility | Mfg. List Price |
|---|---|---|---|---|---|---|---|
| Cardco, Inc. LQ-1 | daisy wheel | 14 cps. | friction | 13 inches | 80 (1 line) characters | parallel & built-in Commodore interface | $649.95 |
| Cardco, Inc. LQ-2 | character drum | 10 cps. | friction | 8.75 inches | 80 (1 line) characters | parallel & built-in Commodore interface | $349.95 |
| Cardco, Inc. LQ-3 | daisy wheel | 13 cps. | friction | 11.5 inches | 80 (1 line) characters | parallel & built-in Commodore interface | $449.95 |
| Epson America, Inc. Epson MX-80 | dot matrix | 80 cps. | friction/ tractor | 10 inches | 1 line | parallel | $495.00 |
| Epson America, Inc. Epson FX-80 | dot matrix | 160 cps. | friction/ tractor | 10 inches | 2K (with serial) | standard: parallel optional: serial | $699.00 |
| Star Micronics Gemini 10X | dot matrix | 120 cps. | friction/ tractor | 10 inches | 4 or 8K | standard: parallel optional: serial | $399.00 |
| Star Micronics Gemini 15X | dot matrix | 120 cps. | friction/ tractor | 15 inches | 8K—expandable to 16K | standard: parallel optional: serial | $549.00 |
| Okidata Microline 80 | dot matrix | 80 cps. | friction (tractor option) | 9.5 inches | — | parallel | $449.00 |
| Okidata Microline 82A | dot matrix | 120 cps. | friction/ tractor | 9.5 inches | 2K (option with serial) | RS-232 and parallel standard; IEEE-488 optional | $549.00 |
| Commodore MPS-801 | dot matrix | 50 cps. | tractor only | 8.5 inches | 1 line | Commodore serial | $279.00 |
| Commodore 1526 | dot matrix | 60 cps. | friction/ tractor | 8.5 inches | 1 line | Commodore serial | $399.95 |
| Commodore DPS-1101 | daisy wheel | 17 cps. | friction | 13 inches | 2K | Commodore serial | $499.95 |
| Commodore MPS-803 | dot matrix | 60 cps. | friction: standard tractor: optional | 8.5 inches | 1 line | Commodore serial | $199.95 |
| Juki Model 6100 | daisy wheel | 17 cps. | friction | 13 inches | 2K | parallel: standard RS-232C: optional | $395.00 |
| Xerox (Diablo) Diablo 630 | daisy wheel | 60-120 cps. | friction/ tractor | 15 inches | 2K | serial or parallel | $1,995 |

# Put the World on Hold!



## Cast off your cares and come sailing in the exotic Caribbean.

Not a dress-for-dinner floating hotel...but "barefoot" sailing & beachcombing for those with adventure in their souls. Lend a hand...or feet on the rail. Six exciting days from $425. Write for your free Great Adventure Book.



**Windjammer 'Barefoot' Cruises**
Box 120, Dept
Miami Beach, FL 33119-0120
or call TOLL FREE (800) 327-2600
in FL (800) 432-3364

Name ____________________
Address ____________________
City ____________________
State/Zip ____________________

**Windjammer 'Barefoot' Cruises**
*Post Office Box 120, Miami Beach, Florida 33139*



# SPACE PATROL FOR THE C-64

BY BOB LLORET

**Captain Microz' unprovoked assault on your world seven years ago laid most of the planet to waste. Since then, you and your fellow survivors could do nothing but grieve and bottle up your rage against your enemy, too powerful to retaliate against with hope of success. But now you've learned through interplanetary informants that the evil Captain has secretly dismantled most of his fleet for retooling. Immediately your forces depart for his world, leaving behind a skeleton patrol of three fighters.**

**What you don't know is that Microz planted the rumor that reached your ears. His ships lurk nearby, ready to finish destroying your world as soon as your armada departs.**

**What Microz doesn't know is that the three ships left behind are experimental FX-100's, equipped with the pinpoint firing accuracy needed to strike the Captain's vessels at their only vulnerable point—the stellar energy cells revolving around their middles. Can your three fighters defeat fifty or more of Microz'?**

*Space Patrol* is an arcade-style game which I consider the best I've produced to date. It incorporates some of the finer features that the C-64 has to offer, such as 3D lettering in the title screen, extensive use of sprites, and screen scrolling for that deep space effect.

Load the program and run it. After looking at the title screen press F1 and the screen will go blank to draw the playfield. After a few seconds the screen goes on and you're ready for action. Use the joystick in Port 2 to make your ship (on the left hand side of the screen) go up or down. The program will control the alien ships. Your score is kept at the lower right hand corner of the screen.

Watch your fuel gauge as you do battle. If you run out of fuel, your ship explodes and the next of three is brought up. To refuel, touch the bottom of the playfield with your fighter.

When your three ships are gone, the game ends and your score is displayed in a computer report, along with a message. If you score over 5000 the message is good; if under 5000, not so good.

I hope you have as much fun playing *Space Patrol* as I had programming it. A copy on disk or cassette can be obtained by sending $6.00 to

Robert Lloret
157 Atlantic Avenue
Staten Island, NY 10304

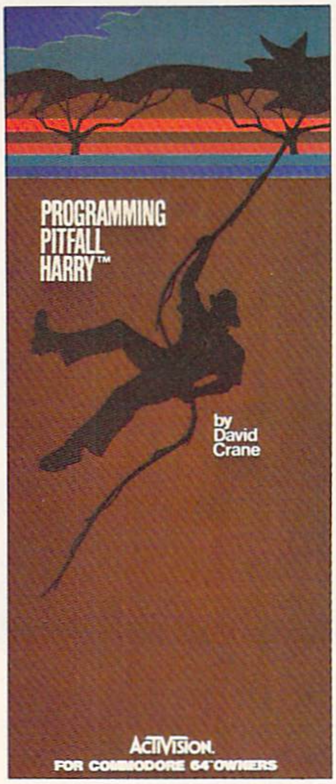
Or you may purchase *Ahoy!*'s monthly disk which has *all* the programs in this issue of *Ahoy!* See page 66 for details. □

*SEE PROGRAM LISTING ON PAGE 79*

# SCUTTLEBUTT

*Continued from page 14*

Each is $19.95 tape, $22.95 disk from Victory Software, Star-Byte, Inc., 2564 Industry Lane, Norristown, PA 19403 (phone: 215-539-4300).



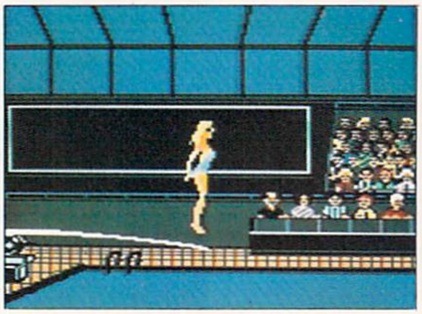

***Program an animated Pitfall Harry.***
*READER SERVICE NO. 13*

Activision is offering C-64 users a chance to put Pitfall Harry on their screens for just 25¢. For that postage and handling fee they'll send a copy of *Programming Pitfall Harry* by *Pitfall* creator David Crane. The booklet contains a short (102-line) program that will generate a moving Pitfall Harry, and instructions for customizing the program by changing the background scenery, the figure's speed and color, and so on.

Tape your quarter to a piece of paper, please, and mail to Activision Commodore 64 Club, P.O. Box 7287, Mountain View, CA 94039 (phone: 415-960-0410).

Activision also informs us that they've contracted to produce a game based on *Ghostbusters*, a film some of you may have seen. More details in a future issue.

*HES Games* requires precise motor coordination to maneuver colorful animated athletes through such Olympic events as weightlifting, diving, running, long jump, archery, and hurdles. You may "instant replay" or save your best performances. For the C-64; $34.95.



**HES Games:** ***practice for 1988.***
*READER SERVICE NO. 18*

HesWare, 150 North Hill Drive, Brisbane, CA 94005 (phone: 415- 468-4111).

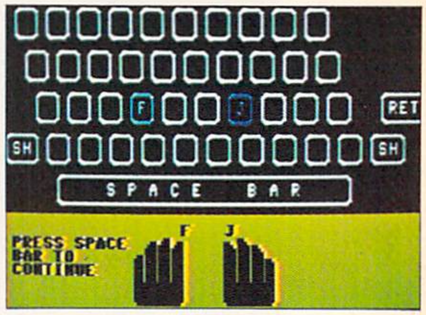
In the latest installment of Infocom's Tales of Adventure series, you're a deep-sea diver who must interact with a band of *Cutthroats* on an expedition for treasure. Price is $39.95 for the C-64 disk and Infocom's usual assortment of maps, books, clues, and artifacts. *Cutthroats* is designated a standard level game, which Infocom describes as a good introductory level for adults.

An even better introduction to text adventuring might be Infocom's new Four-in-One Sampler, providing excerpts from *The Witness, Infidel, Planetfall,* and *Zork I,* along with a tutorial on interactive fiction. Included is an $8 rebate coupon on the purchase of any full-length Infocom title.

A 16-page catalog of all Infocom's games and accessories is available free at many dealers.

Infocom, Inc., 55 Wheeler Street, Cambridge, MA 02138 (phone: 617-492-1031).

## *HUNT AND PEEK*




**Type 'N' Write:** ***for computer users.***
*READER SERVICE NO. 19*

Just typing the *names* of all the typing tutorials out for the C-64 would be practice enough to increase most people's speed by 10 words per minute. Into this glutted marketplace comes HesWare's *Type 'N' Write,* a program with a twist: it's directed toward the computer, not the typewriter, keyboard.

Instruction is provided in drill and touch typing, as well as in word processing and computer functions like word wrap, insert/delete, cursor movement, saving, retrieving, and listing files, and 40/80 column printing. Typing speed, words per minute, and number of errors are displayed onscreen. Included is a word processor that can create up to eight pages of text. C-64 cartridge; $29.95.

HesWare, 150 North Hill Drive, Brisbane, CA 94005 (phone: 415-468-4111).

Two more trivia game releases added to last issue's list:

*Factactics* includes 2000 questions on music, movies, sports, and Americana. Players progress around an onscreen gameboard by

supplying the missing fact in detailed statements via fill-in, multiple choice, or true/false. For the 64; $29.95.

Daystar Learning Corporation, 525 University Avenue, Palo Alto, CA 94301 (phone: 415-323-3567).

*Trivia Mania* offers C-64 users 1000 questions in 6 categories and four levels of timed play. Diskette or board game is $39.95.

Dynacomp, Inc., 1427 Monroe Avenue, Rochester, NY 14618 (phone: 716-442-8960).

## CREATING YOUR Own GAMES

*Continued from page 54*

out your own designs, just replace any of the 8-number patterns with some of your own. As long as you always have eight numbers after the letter you want to replace in the DATA statement, your pattern will take the place of the character that begins the DATA statement.

The Commodore 64 allows you to do even fancier things with your custom characters. Next month, we'll do some character-flipping—a technique that lets you do extremely fast animation with only a few POKEs. You not only can make pictures with characters—you can also make them move. □

***SEE PROGRAM LISTING ON PAGE 75***

## COMMODARES

*Continued from page 56*

ans). Mr. Steinberg also sent along a mnemonic he uses to remember the value of pi:

See, I have a rhyme assisting
My feeble brain, its task sometime resisting.

The number of letters in each word corresponds to a digit in pi.

As usual, there are many more programs and ideas from all of you that wrote than we have room to print. Don't let that discourage you from writing and sharing your thoughts, however. See you next month with more challenges. □

## RUPERT REPORT

*Continued from page 91*

to which line A branches.

Line 10250 defines the largest line number that is to be cross-referenced. By setting MX to 9999, this program will stop before analyzing itself. The first lines of the program in this listing are just to show the types of branch statements that are possible. The program you want to analyze will replace these lines. The first line of the program should be GOTO 10000 in order to jump to the cross reference program.

You also have the choice of sending the output to the screen or printer. You should change the file-number in line 10370 to a number less than 128 if your printer does not require a line to feed after each carriage return. The token codes in line 10270 were determined from the program discussed earlier.

This program steps its way through memory byte by byte, looking for keywords corresponding to branch statements. As written, it will not recognize GO TO if they are separated. Notice from our earlier program that GO, TO, and GOTO each have separate tokens. If you normally leave a space between GO and TO, or if you just want to study this program, you might modify it to handle GO TO.

If you are analyzing a long program, you may have to increase the size of the array in line 10240. As dimensioned, it allows for 200 program lines and a maximum of ten branches within each line. You may change those values to suit your needs.

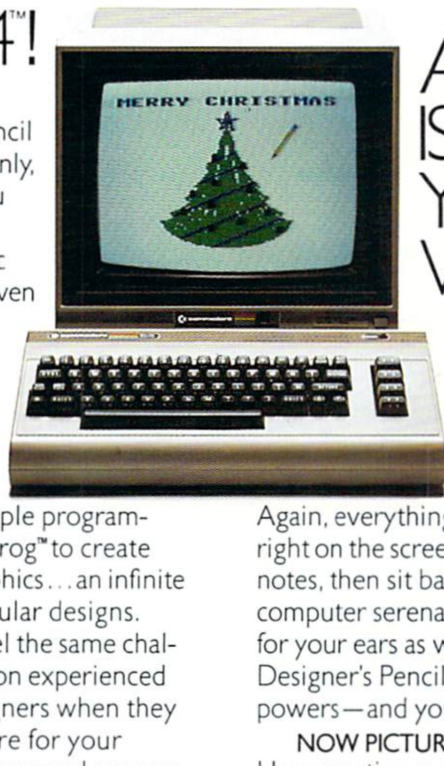
The executive program is in lines 10010 through 10099. The comments and previously discussed concepts should make this program understandable.

We have learned how BASIC programs are stored and may be analyzed. With this information you should be able to do some serious investigation, not to mention some tricky programming. You should now be able to see what is involved in a BASIC renumbering program. Maybe you'd like to write one of your own. Other possibilities might be a variable cross-referencing program or a utility to selectively delete or rearrange line numbers. Dive in! The fun begins when you start exploring. □

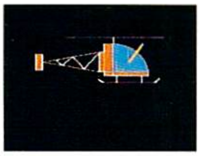
***SEE PROGRAM LISTING ON PAGE 73***

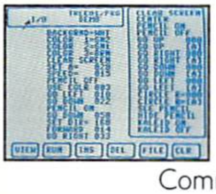
# SUDDENLY,

## YOU CAN DRAW ANYTHING ON YOUR COMMODORE 64™!




Put The Designer's Pencil in your hands. And suddenly, using only a joystick, you can command all of your computer's graphic abilities. You don't even have to be able to draw, because the computer does it for you. And The Designer's Pencil doesn't just doodle around. It uses a revolutionary, simple programming technique called Prog™ to create actual computerized graphics . . . an infinite number of spectacular designs.

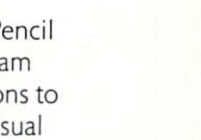




You'll be able to feel the same challenge and satisfaction experienced by Activision designers when they create new software for your Commodore 64. Every command appears right on the screen as shown here. Just choose what you want to do, then watch as the computer carries out your every wish.

## AND IF THAT ISN'T MUSIC TO YOUR EARS, WRITE A TUNE.

The Designer's Pencil also lets you program musical compositions to accompany your visual masterpieces.




Again, everything you need appears right on the screen—just choose your notes, then sit back and let your computer serenade you. It's a delight for your ears as well as your eyes. The Designer's Pencil will amaze you with its powers—and yours.

NOW PICTURE YOURSELF A WINNER. How creative can you be? We can't wait to see. Use The Designer's Pencil to create your wildest fantasies, then enter the results in The Designer's Pencil $10,000 Contest. Details in every specially-marked package.



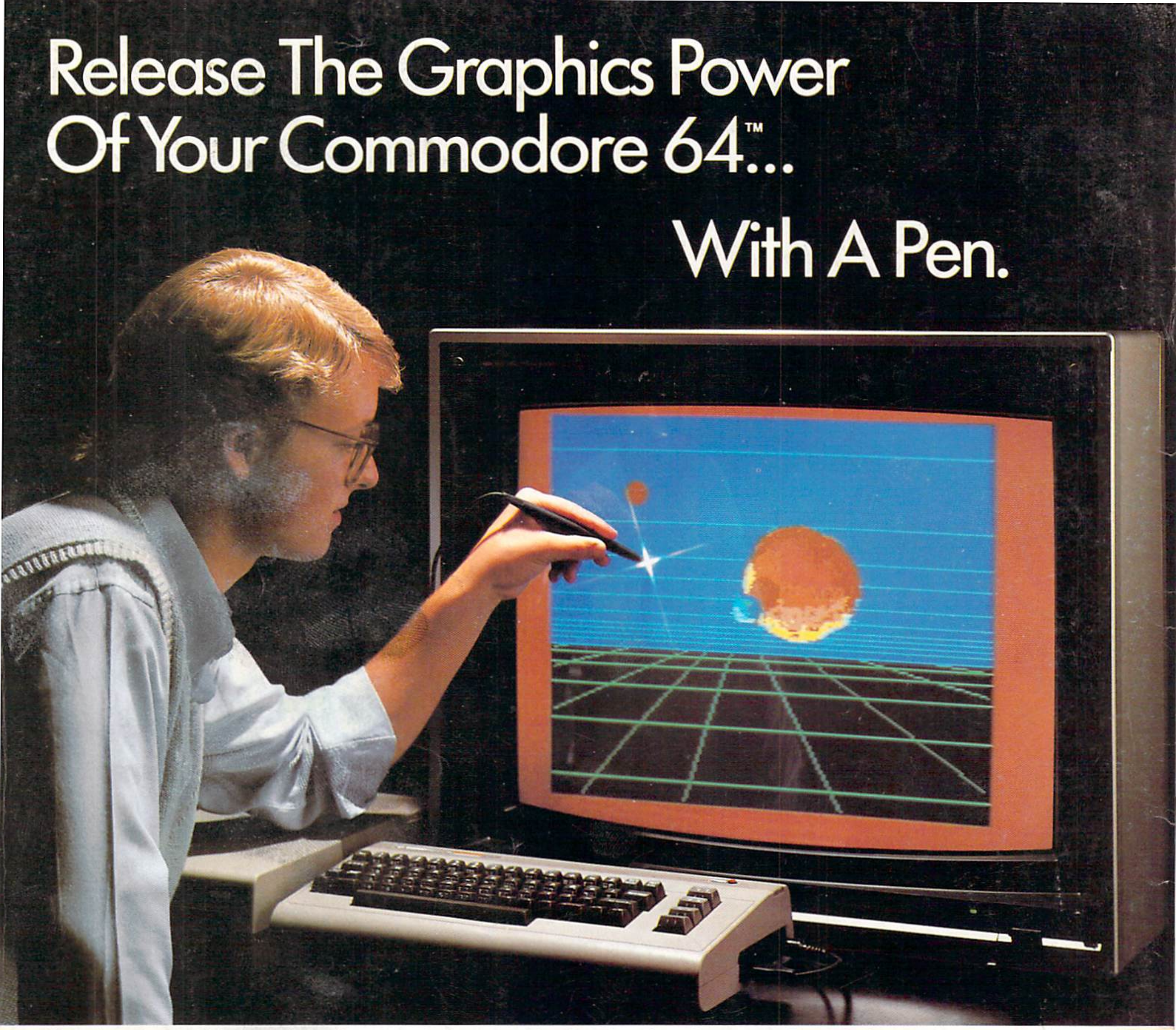

Designed by Garry Kitchen.

**THE DESIGNER'S PENCIL™**

**ACTIVISION®**

# Release The Graphics Power Of Your Commodore 64™...

# With A Pen.



Now there's a sophisticated graphics system for your C-64—the Koala Light Pen System™ from Koala Technologies. It's easy. You work directly on your computer screen to draw, design, chart graphs. The Koala Light Pen System is all you need.

## A State-Of-The-Art Light Pen

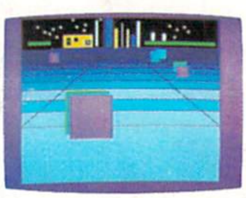
We've packed all the electronics in the streamlined Koala Pen, using the most advanced miniaturization techniques.




This means clearer, smoother, more precise graphics—and, unlike other C-64 graphics products, you can use the full range of the C-64's colors. It also has an easy-to-use tip-switch.

## A Complete Graphics Package

Koala, well known for our graphics systems, has created the premier



C-64 graphics package. It contains a complete array of graphics capabilities. In addition, you can dip into a palette of 16 colors—and over 3,000 textured patterns. Point to "mirror" and a kaleidoscope of colorful images bursts onto the screen. Zoom in for 320 x 200 resolution. "Rubber band" lines and shapes.

Explore the full graphics power of your C-64 with the Koala Light Pen System. There is no other system like it.

## The Koala Light Pen System™ and Graphics 64 Software.™



Koala Technologies Corporation
800-KOA-BEAR

Reader Service No. 54